BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

(19) Europäisches Patentamt
European Patent Office
Office uropéen des brevets

(11) Publication number: **0 180 258**
A1

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 85201527.0

(22) Date of filing: 24.09.85

(51) Int. Cl.⁴: G 06 K 15/02, G 06 K 15/12

(30) Priority: 02.10.84 NL 8402998

(43) Date of publication of application: 07.05.86 Bulletin 86/19

(84) Designated Contracting States: DE FR GB IT NL

(71) Applicant: Océ-Nederland B.V., St. Urbanusweg 43, NL-5914 CC Venlo (NL)

(72) Inventor: Peters, Hendrikus, Laan ten Boomen 9A, NL-5715 AA Lierop (NL)
Inventor: Jongen, Alphons, Bergerweg 26, NL-6063 BR Vlodrop (NL)
Inventor: Tunnissen, Franciscus, Frederik Hendrikstraat 57, NL-5923 BK Venlo (NL)

(74) Representative: Bleukx, Lucas Lodewijk Maria, Ir. et al, Océ-Nederland B.V. Patents & Information Dept. Postbus 101, NL-5900 MA Venlo (NL)

(54) Decoder.

(57) Decoder for converting information supplied in code about a compiled page into image formation which is fed in the form of a serial pixel bit flow to a raster output scanner, in which the raster output scanner is designed to form with the aid of a number of scanning lines an image on an image-forming medium, comprising a first memory for the storage of all items occurring on the compiled page together with the m-bits information relating to their position on that page, a second memory for the storage of the bit pattern of all items occurring in the first memory, a third memory for the storage of the address information relating to the bit patterns in the second memory and for the storage of format information for items, means for converting data taken from these tables into n-bits words for each scanning line, and at least on n-bits wide pixel column memory.

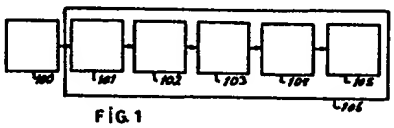
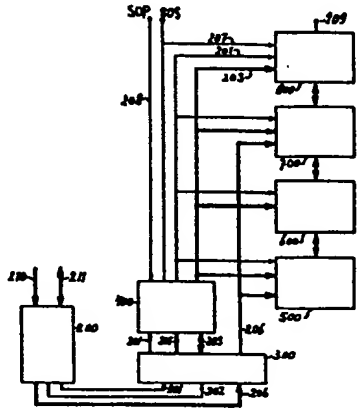


FIG. 1



EP 0 180 258 A1

0 180 258

Océ-Nederland B.V., at Venlo

Decoder

This invention relates to a decoder for converting information supplied in code about a compiled page into image information which is fed in the form of a serial pixel bit flow to a raster output scanner, in which the raster output scanner is designed to form with the aid of a number of scanning lines an image on an image-forming medium, comprising a first memory for the storage of all items occurring on the compiled page together with the m-bits information relating to their position on that page, the sorted table of items (STI), a second memory for the storage of the bit patterns of all items occurring in the first memory, the pixel pattern data table (PPD), a third memory for the storage of the address information relating to the bit patterns in the second memory and for the storage of format information for items, the table of pointers to pixel pattern data (TPPD), means for converting data taken from the sorted table of items, the pixel pattern data table and the table of pointers to pixel pattern data into n-bits words for each scanning line, and at least one n-bits wide pixel column memory.

Such a decoder, specifically suited for a xerographic laser printing system, is described in the United States patent No. 4 079 458. In that patent, data relating to the characters to be printed on lines are sorted in such a way that all characters which start on one and the same scanning line are grouped together. The data of the characters which start on successive scanning lines are stored in an input memory, and the data relating to partially processed characters are stored in an active memory. Pixel pattern data for a number of characters are stored at separately accessible address locations in a font memory and these data are retrieved as a function of the data from the input memory and the activity memory during the successive scanning lines. The retrieved data are placed in an output memory and are then fed in the form of a serial pixel bit flow to a xerographic printer. To increase the processing speed, two output memories have been fitted. Whilst data are being read out of the first output memory, the second output memory can be filled.

None the less, no very fast processing speed can be achieved using the known decoder. The displacement of the word data in c nformity with their r levant address s and also the design of the output memories themselv s are not conduciv to fast d coding.

The object of the invention is therefore to provide a decoder in which these drawbacks are eliminated.

In the case of a decoder as described in the introduction, this object is achieved in that the decoder comprises:

- rotation logic, designed to shift the n-bits words in a register as a function of the M least significant bits of the m-bits information about the position of the items on the page as stored in the first memory, whereby $2^M = n$ applies,
- a circuit, designed to store the information not shifted out of the register of the rotation logic in the pixel column memory at an address location which is double that of the (m - M) most significant bits of the said position information, and to store the information shifted out of the register at the next-higher address location,
- a first parallel-in-serial-out shift register which can be connected via a buffer to the odd address locations of the pixel column memory,
- a second parallel-in-serial-out shift register which, in relation to the first shift register, can be connected to the next-higher address locations of the pixel column memory,
- a counter connected to the address lines of the pixel column memory for generating successive addresses when that memory is being read out, and
- a combination circuit linked to the serial-out connections of the first and the second shift registers.

In the entire decoder parallel data processing is applied so that the conversion of information supplied in code into the required image information can take place especially rapidly. Moreover, it is possibl to make do with little memory capacity.

Other features and advantages will be made clear in the following description, in which reference is made to the attached drawings, of which:

Fig. 1 is a schematic representation of the location of the decoder in an image processing station,

Fig. 2 is a schematic representation of the decoder,

Fig. 3 is a block diagram of the item pr cessor of the decoder,

Fig. 4 is a block diagram of the item-slic proc ssor f the d coder,

Fig. 5 represents the structure of a pixel pattern of an item,

Fig. 6 is a block diagram of the pattern data generator of the decoder,

Fig. 7 is a block diagram of the column data generator of the decoder, and

Fig. 8 is a block diagram of a barrel-shifter.

On an input station 100 Fig. 1 (word processor) a page is typed in and sent to the printer 106. In the printer 106 the incoming page is converted with the aid of translator 101 into a code suitable for the printer, the UIL (universal input language). This information in UIL code is fed to the formattor 102 in which the formatted page is compiled. This contains each character (or font) with its X and Y positions on the page and its font code. The formatted-page information is now arranged in the converter 103 to make what is known as a converted page, suitable for the decoder 104. The decoder 104 translates the arranged information into a video signal which is used to modulate the laser beam of the laser printer 105. With the aid of this laser beam an electrophotographic image is written in accordance with the known method and the laser printer 105 prints out this image on plain paper. The printer which is used with the decoder according to the invention is designated to process a page in the landscape form as well as in the portrait form, but in order to give a clear description of the printer it has been assumed that a page of text in the portrait form is supplied from a word processor. The laser printer reproduces this information rotated through 90 degrees. The terms X and Y direction relate to the information originally supplied, in other words the X direction is the direction parallel to the lines and the Y direction or SCAN direction is the direction at right-angles to the lines.

Via the converter 103, the decoder 104 receives three different tables, known as the decoder tables. The first table is called the "sorted table of items" (STI) and contains all items which occur on a page in their sorted X sequence, whereby an item is understood to mean a letter, number or logo. All items which start with a same X value are group d and a s t of items with the same X value is referr d to as a column. All columns are closed with an "end of column"

(EOC) bit. This EOC bit is also present wher a column remains empty if not a single item starts with the relevant X value. If printing has to be done at a resolution of 12 dots/mm, about 2520 columns are needed for one A4 size page. This means that the STI table in fact consists of 2520 sub-tables of varying lengths. If a lot of items start with one specific X value, such a sub-table will be long (for example, 600 locations), whilst a sub-table for an empty column will consist of one location.

Each element of the STI table contains an item number (this is a unique code for each character or logo) and the Y starting value for that item on the page to be printed. Another table of the decoder tables is the "Table of Pointers to Pixel Pattern Data" (TPPD), which is a vector table. The vector is an address in the pixel pattern memory. In this tabel the relevant information on the height and width of the item is also stored for each item number. The height of an item is indicated in units of 16 pixels, the width in units of 1 pixel. The last table of the decoder tables is the "Pixel Pattern Data" table (PPD) which contains the pixel pattern of all items occurring on the page and which can best be described as a sort of loadable character memory.

The items are built up of pixel words of 16 pixels. The collection of all 16-bits pixel words in one item, sorted according to their ascending Y position and with one value for X, is referred to as an item slice.

Once the decoder has been loaded with the above-mentioned tables, the decoding process can start. One column is read out of the STI table each time and each item occurring in this is expanded with the aid of the TPPD table and the pixels are supplied to a line buffer. If an item in a preceding scan has still not been completely processed, the remaining information from it is placed in a tally memory. Each element in this tally memory contains the information on the height of the item, the width that remains to be processed, the Y position and the address in the PPD table where the relevant pixel pattern of the item being processed is located.

Each element from the tally memory therefore comprises two types of information, viz. information which makes it possible to address the memory in which the PPD table is located, and information relating to

the Y-position which indicates the starting point from which the relevant pixels for this item should be placed on paper. During each scan (Y cross-section of the page) all elements of the tally memory are processed and, if an item has still not been fully processed, it is put back in the tally memory. If all elements of a scan have been processed, new elements of a following column are processed and, here again, the still not fully processed items are put back in the tally memory. All pixels of a scan are read into a first line buffer and are read out in response to a command from the laser printer whilst a second buffer line is being filled with the information for the following scan.

Fig. 2 gives a schematic representation of the decoder. The actual decoder part consists of the item processor 500, the item-slice processor 600, the pattern data generator 700 and the column data generator 800. For controlling purposes a monitor 200, page-control 400 and a bus-interface 300 have also been added.

The information from a page is passed via bus 210 to monitor 200, whilst the decoder dialogue between an input station and the monitor 200 is transmitted via bus 211. The system-clock line 201 is linked to the other modules via bus-interface 300.

Controlling signals, such as the "decode-a-page" signal, are linked to the various modules via bus-interface 300 through bus 203. The information on the contents of a page is supplied to the modules via bus 206 and bus-interface 300.

The function of page-control 400 is page synchronisation: synchronising the laser printer and the decoder with a "decode-a-page" command.

The item processor 500 comprises the memories for the STI and TPPD tables. There are loaded via bus 206. The item processor 500 also supplies items in response to requests from the item-slice processor 600.

The item-slice processor 600 receives items from the item processor 500 and supplies item-slices in response to requests from the pattern data generator 700. Items which consist of several slices are temporarily stored in a tally memory in the item-slice processor 600.

The pattern data generator 700 receives item-slices from the item-slice processor 600. Via bus 206 the 16-bit pixel patterns are retrieved across their full height from the PPD memory and are supplied, together

with their relevant Y positions, to the column data gen rator 800 upon request. The column data generator 800 receives the pixel patterns and the Y position f an item via pattern data generator 700. For the laser printer a pixel bit flow 209 is generated, which supplies the pixel patterns serially according to their Y sequence. Pixel patterns with an identical Y position (overlay) are combined. A "start-of-page" signal (SOP) is fed from the laser printer via line 208 to page-control 400. The "start-of-scan" signal (SOS) is fed via line 207 to page-control 400 and to the column data generator.

Item processor

The operation of the item processor will be explained by reference to Fig. 3. Via buffer 502 the addresses of the items occurring on a page which are supplied on bus 501 are fed to STI & STI Control block 503 as well as to TPPD & TPPD Control block 504. Via buffer 506 the corresponding data supplied on bus 505 are likewise fed to block 503 and block 504. Buses 501 and 505 form part of bus 206 as indicated in Fig. 2. Via buffer 509 the internal control bus 512 is linked to bus 203.

Blocks 503 and 504 both comprise a memory and a control section. In a period when no "decode" signal is present on bus 512, the memories in blocks 503 and 504 are filled with information about a compiled page. The STI table is read into the memory of block 503 and the TPPD table into the memory of block 504. The STI table is a 24-bits wide table which is subdivided into 12 bits which indicate the item's Y position on the page, 10 bits for the item number, one "end of column" bit (EOC) and one reverse bit. If a "decode" signal is supplied to bus 512, then the information relating to the items occurring on a page will become available item by item from the STI memory via buses 520 and 525. The 10 bits which represent the item number are supplied via bus 520 to the TPPD & TPPD control block 504. They constitute the address for the TPPD table which is present in the memory of block 504. The remaining 14 bits of the STI table are fed via bus 525 to an output register 507.

The TPPD table which is present in block 504 is 30 bits wide. Of these bits, 18 bits indicate the PPD base address whilst, of the remaining 12 bits, 8 bits represent the width of the item and 4 bits represent the height of the item.

During the dec d state these data are likewise supplied from the TPPD table via bus 521 to th output register 507. Bus 522, which c nnects the output register 507 with the item-slice processor 600 and which is 44 bits wide, thus holds all information relating to one item, viz. the Y position on the page (12 bits), the PPD base address (18 bits), the width (8 bits) and the height (4 bits), the EOC bit and one reserve bit.

The item processor 500 only supplies data to the item-slice processor 600 at the latter's request. For this purpose a data-request signal 526 is fed to output control 508. The item processor transmits a data-available signal 527 to the item-slice processor if the requested data are present in output register 507.

A counter 510 is linked to the STI & STI Control block 503. The counter readings are used to read out the STI table sequentially, whilst the counter reading indicates the address in the table. When a decode-a-page signal is sent via the item-slice processor 600 to output control 508, the latter generates a counter-reset signal which is supplied to the counter via line 528. The read-out of the STI table starts at address 0. At every data-request signal 526, the counter reading is increased by one via line 529.

Item-slice processor

Fig. 4 is a schematic representation of the item-slice processor (ISP). During the decoding process the item-slice-processor control 609 (ISP control) generates a data-request signal 526 for the item processor 500. If this item processor is ready for data transmission, it gives a data-available signal 527 to the ISP control 609. Then one column is retrieved from the item processor 500 through bus 522 and via input register 601 and this column is passed on to the pattern data generator 700 via output register 602 and bus 617. One column contains all items having the same value of X which occur on a page.

The 44 bits wide bus 522 is divided up in the ISP into a 35 bits wide bus 612, which transports the PPD base address, the height of the item and its Y position, a line 613 through which an EOC bit can be transported, and an 8 bits wide bus 614 which transports information relating to the width-1 of an item.

If an item is composed of mor than one slice (which will usually be the case), the data of that item are stored via register 604 into the

tally memory 605. This tally memory is a 2K x 48 bit memory. The bit information relating to the width of the item is r duced by one with the aid of a subtraction circuit 610 and is also stored in the tally memory 605. The portion of an item that is left over after one or more slices have already been removed is referred to as the item remainder. If a first column has been sent in this way via bus 617 to the pattern data generator 700, which is indicated by an EOC bit, then a start can be made on the second column.

For this purpose the item remainder of the first item is supplied from the tally memory 605 via buffer 603 and output register 602 through bus 617 to the pattern data generator 700. The W=0 detector 611 serves to ascertain whether the width-1 information relating to the item or item remainder being processed has decreased to zero. If this is not the case, the bit information relating to the width of the item is again reduced by one with the aid of the subtraction circuit 610. This new item remainder is stored in the tally memory 605 again. If the width-1 information has decreased to zero, then the item has been fully processed and is therefore no longer fed via register 604 to the tally memory.

After all items stored during the previous column have been processed in the tally memory 605, the new items which belong to this column are retrieved from the item processor and dealt with as described above. To enable effective use to be made of the tally memory 605, two counters 607 and 608 as well as an address selection block 606 have been added.

During writing to the tally memory 605, this memory is addressed with the aid of write-counter 608 via the address selection block 606. At the start of the reading-in of the first column from the item processor 500, write-counter 608 is reset to zero. The first item remainder fed to the tally memory 605 is thus written to address zero.

Subsequently, the counter reading of write-counter 608 is increased by one so that the second item remainder from the first column is written to address one.

All item remainders from the first column are then written successiv ly by ascending address into the tally memory 605. The last item remainder is placed at address N. After that, an EOC bit which originates from the item processor is st red at addr ss N + 1.

When th s cond column is being processed, read-c unter 607 and write-counter 608 are first res t to zer . The item remainders in the tally memory 605 are read out in conformity with the address indicated by counter 607. In other words, the item remainder at address zero is read out first. If a new item remainder has again been left over from this item remainder, such remainder will again be written to address zero, as indicated by write-counter 608.

The read-counter 607 is increased by one and the following item remainder is read out. The write-counter 608 is only increased if a remainder has again been left over from the item or item remainder which is being processed. In this way the tally memory 605 is always filled from address zero onwards. The read-out of the tally memory 605 continues until the EOC bit is read. After that, new items forming part of the columns being processed are retrieved from the item processor 500, are processed, and any item remainders are successively stored in the tally memory 605 until an EOC bit is received again. Item-slice-processor control 609 consists of a Field Programmable Logic Array (FPLA) and a register, which together constitute a "State machine". The FPLA determines the new state on the basis of the information supplied to the inputs and the state which is being processed. The information emerging at the output simultaneously determines the new state being processed.

The operation of the various blocks of the item-slice processor is controlled by the item-slice-processor control 609. The control communication between the item-slice processor and the other parts of the decoder takes place via bus 612. The data-available signal 527 from the item processor and the data-request signal 615 from the pattern data generator are linked with item-slice-processor control 609, as are the data-request signal 526 and the data-available signal 616.

The pattern data generator

The principal task of the pattern data generator is to translate the item-slice information supplied by the item-slice processor into pixel patterns. The item-slice information comprises the data relating to the height of the item, the Y position on the page, the base address of the PPD memory at which the pixel information for the relevant item has been stored, and the item-progress. The item-progress is underst od t mean th it m remaind r width-1 information. After being processed

in the pattern data generator, the 16-bits words which repr sent th pixels in a slice are supplied, together with their r levant Y positions, to the column data generator 800 in respons to its requ st.

Fig. 5 represents how the pixel patterns of an item are arranged in the PPD memory. The example used is a letter N (763), which has been divided into a matrix of 11 x 5 sectors. The item-progress has been plotted along horizontal axis 765, whilst the height minus 1 in words (=H) has been plotted along the vertical axis 766. The starting dot of the letter is sector 764. When defining the X and Y position of this item on the page, the position is related to this sector 764. The base address of this item is the address of memory location 750 where the pixel pattern of sector 764 is to be found.

An item-slice with 5 consecutive 16-bits words is indicated by the numbers 758 to 762 inclusive. The item-progress indicated on axis 765 is four for this slice. The item-slice word 758 is placed in the memory at location 751. The word 759 is at memory location 752 and word 762 is at memory location 755.

The address of memory location 751 corresponds to the base address 750 plus the product of the height minus 1 of the item + 1 (= H + 1) and the item-progress. If the base address 750 of the PPD memory were 1000 (hex$), then the address of word 758 and hence of memory location 751 in the PPD memory would be 1015 (hex$).

In Fig. 6 bus 617 is the connection between the output register 602 of the item-slice processor 600 and the input register 701 of the pattern data generator 700. This 44-bits-wide bus 617 is divided up in such way that the offset calculation block 702 receives the bit pattern forming part of the item-progress (Ip) (8 bits) via bus 717 and the bit pattern forming part of the item height (H + 1) (4 bits) via bus 718. In block 702 the product of the item-progress (Ip) and the height (H + 1) is calculated (Ip x (H + 1)). The offset signal thus calculated (12 bits) is sent through bus 719 to the offset address counter 705. In summing circuit 708 the offset address is added to the base address (18 bits), which is supplied via bus 716, register 704 and bus 722. Bus 723 is thus given the address of the first word of the relevant item-slice in the PPD memory 710.

Subsequently, the height of the item-slice being processed is counted down to zero in height-counter 706. F r each subtraction step

of the height-counter 706 th c nt nts of the offset address counter 705 are increas d by one and this increased value is in turn added to the base address in summing circuit 708. In this way the consecutive addresses which form part of a specific item-slice are selected in the PPD memory.

The 12-bits-wide information supplied through bus 617 relating to the Y position of an item on a page is divided up into the 8 most significant bits, which are fed via bus 713 to the Y-destination counter 703, and the 4 least significant bits, which are fed via bus 714 to register 704. These four least significant bits of the Y position represent the shift information. Via bus 714 and pipeline register 709, this shift information is supplied via output register 711 to bus 712.

For each subtraction step of the height-counter 706, the contents of the Y-destination counter 703 are also increased by one. This new Y-position value is sent through bus 727 to output bus 712.

The pixel pattern which belongs to the address supplied on bus 723 is fed to the column data generator 800 via the output register 711 through bus 712.

The pattern-data-generator-control block 707 ensures the correct operation of the generator with the aid of control signals which are sent to the various counters and registers. Via bus 728 this control block 707 communicates with other control blocks in the decoder (clock signals, decode signal, etc.). A data-available line 725 and a data-request line 726 have been installed between the column data generator 800 and the pattern data generator 700.

The column data generator

The task of the column data generator 800 (Fig. 7) is to convert the (parallel) information that has been supplied into a serial pixel bit flow for the laser printer. Via bus 712 the pipeline register 802 of the pattern data generator 700 receives the pixel bit pattern of the items (16-bits words) and also the relevant Y positions in respect of one specific X position. At this point the sequence is still arbitrary. These pixel bit patterns have to be sorted according to their ascending Y position, whilst pixel bit patt rns with an identical Y positi n (ov rlays) have to be combined.

Since rapid signal processing is essential, the pixel column memories

812 and 813 are designed as separate 16-bits-wid and 512-words-deep memories. Whilst one pixel column m mory is b ing read out, the other pix l column memory is being filled with 16-bits words. To make it possible to address the 16-bits words in the pixel column, use is made of the 8 most significant bits of the Y position. So that it is still possible to indicate the position of an item on a page with an accuracy of 1 pixel whilst not having to abandon the simultaneous processing of 16-bits words, the four least significant bits, prior to their being stored in one of the pixel column memories 812 and 813, are used to ensure that the bits in the separate 16-bits words are displaced in such a way that the starting dot of an item (the word border) corresponds to the required Y position.

This displacement is effectuated in a rotation logic block 806 with the aid of a barrel-shifter. The operation of this barrel-shifter 806 will be explained by reference to Fig. 8.

Via data-bus 824 the 16-bits-wide words, which represent the pixel pattern of the item being processed are fed from the PPD memory to the input register 825 of the barrel-shifter 806.

Of the 12-bits information relating to the Y position of that pattern on the page, the 4 least significant bits are separated off and are also fed to barrel-shifter 806 via address bus 804. Depending on the addresss information on bus 804, the information which is present in input register 825 is shifted and stored in output register 827. Input register 825 and output register 827 form part of pipeline register 802 and pipeline register 808 respectively.

In input register 825 the letters a to p inclusive represent the contents of the register locations which are coupled to the respective data lines DO and D15 inclusive. If it is assumed that at address bus 804 (via address lines A0 to A3) the least significant bits of the Y position correspond to, say, binary 3 (= 0011), then the information from input register 825, shifted by three places, will be fed into the output register 827. The information that has been shifted out (a,b and c) will again be stored at the front in output register 827.

To enable high-speed processing, the barrel-shifter 806 is designed in such a way that shifting is effected in parallel, on all 16 bits at the same time.

The pixel column mem ries 812 and 813 are each composed of 16 memories f 512 x 1 bit which are us d in parallel for writing-in of 16-bits words. These memories are addressed by using the 8 most significant bits of the Y position (the Y-word address), which are fed to the addresses A1 - A8 of the 16 memories via pipeline register 802, bus 803 and pipeline register 808 through bus 810. The A0 address line of each memory IC is linked via bus 811 to A0 generation block 805.

The A0 generation block 805 has an input bus 804 through which the 4 least significant bits of the Y position are supplied. Depending on these inputs signals, one or more of the 16 outputs, each of which is linked to the A0 address of one memory IC of memory 812 or 813, may become a logical 1 or 0. For instance, if a signal supplied to the input corresponds to binary 3 (= 0011), then three output lines which are linked to the A0 address lines of three memory ICs from pixel column memory 812 or 813 will become a logical 1, which means that these memories will be addressed at the next-higher word address compared to the other 13 memories. This makes it possible for the information shifted out in rotation logic block 806 (for example, bits a, b and c in Fig. 8) to be written to the next-higher word address in the pixel column memory 812 or 813. The overlay-logic block 809 comprises a register to which the data are supplied from the output register of barrel-shifter 806, which is situated in pipeline register 808. The pixel column memories 812 and 813 are addressed using the word address pertaining to those data, whilst the data already present at that address are also fed via bus 831 to overlay-logic blok 809. These data are combined via an OR relation with the data resident in the overlay register, and the result of this is written back again to the selected address.

If one pixel column memory has been completely filled with the information from a scanning line, this memory is switched to the read-out state, whilst the other memory is changed from the read-out state into the write-state. Output control 814 gives the command for this to the pixel column memories via bus 833. During the time that one memory is being filled, the other memory is read out.

Output bus 815 of the pixel column memori s 812 and 813 is linked to two r gist rs 816 and 817, ach of which is on word wide and each of which is connected through t a 16-bits shift register, 818 and 819

respectiv ly. Output control 814 supplies the output addresses A0 to A8 inclusiv to th pix l column memory 812 or 813 with the aid of a counter and via bus 833. During this read-out all A0 address lines of the 16 memory ICs are interlinked. The 16-bits data word at address location 1 of the pixel column memory is transmitted in parallel via bus 815 to register 817. Then, address location 1 is cleared (set to zero) and the address is increased by one, with the result that address location 2 is selected. The data at address location 2 are supplied to register 816 via output bus 815. Address location 2 is now cleared. In this same manner the shifted bits of a word, which had been stored at an even address location are supplied each time to shift register 818, whilst the non-shifted bits of the following word, which had been stored at an odd address location, are supplied to shift register 819.

The laser printer generates an internal burst signal which corresponds to the required write-frequency (the pixel-bit rate) of the pixel bit flow to be fed to the printer. This burst signal is transmitted via line 829 to output control 814 and is then distributed to shift registers 818 and 819 and to flip-flop 828. By means of this burst signal both the shift registers 818 and 819 are serially clocked out at the same time and the data from these registers are combined in combination circuit 820.

The output of combination circuit 820 is transmitted to flip-flop 828 and the burst signal ensures that the pixel bit flow to the laser printer via line 821 is synchronised when it leaves the column data generator 800 via line 821.

The control of the input logic of the column data generator is handled by input control 801. Data-request line 726 and data-available line 725 are also connected to input control 801. The EOC bit is also fed to input control 801 via line 834.

Although only one preferred embodiment of a decoder according to the invention has been described, it will be clear that modifications to this embodiment are also possible, which modifications will fall within the scope of the invention as specified in more detail in the claims set forth below.

CLAIMS

1. Decoder for converting information supplied in code about a compil d page into imag information which is fed in the form of a serial pixel bit flow to a raster output scanner, in which the raster output scanner is designed to form with the aid of a number of scanning lines an image on an image-forming medium, comprising a first memory for the storage of all items occurring on the compiled page together with the m-bits information relating to their position on that page, the sorted table of items, a second memory for the storage of the bit patterns of all items occurring in the first memory, the pixel pattern data table, a third memory for the storage of the address information relating to the bit patterns in the second memory and for the storage of format information for items, the table of pointers to pixel pattern data, means for converting data taken from the sorted table of items, the pixel pattern data table and the table of pointers to pixel pattern data into n-bits words for each scanning line, and at least one n-bits wide pixel column memory, characterised in that the decoder comprises rotation logic (806), designed to shift the n-bits words in a register (827) as a function of the M least significant bits of the m-bits information about the position of the items on the page as stored in the first memory (503), whereby $2^M = n$ applies, a circuit (AO generation-block) (805) designed to store the information not shifted out (a-c) of the register of the rotation logic (806) in the pixel column memory (812, 813) at an address location which is double that of the (m - M) most significant bits of the said position information, and to store the information shifted out (d-p) of the register (827) at the next-higher address location, a first parallel-in-serial-out shift register (818) which can be connected via a buffer (816) to the odd address locations of the pixel column memory (812, 813), a second parallel-in-serial-out shift register (819) which, in relation to the first shift register (818), can be connected to the next-higher address locations of the pixel column memory (812, 813) a counter connected to the address starting points of the pixel column memory (812, 813) for generating successive addresses when that memory is being read out, and a combination circuit (820) linked to the serial-out connections of the first (818) and the sec nd (819) shift registers.

2. Decoder according t claim 1, characterised in that the rotation l gic (806) comprises a barrel-shifter which is connected to an n-

bits input register (825), an n-bits output register (827), and a circuit which, as a functi n f M control signals, can shift the data present in the input register (825) and place them in the ouput register (827).

3. Decoder according to claims 1 or 2, characterised in that the pixel column memory (812,813) is composed of n memory elements of which from each memory element one adress line is linked to one of the m - M outputs of an AO generation block (805) and of which the remaining address lines are linked together.

4. Decoder according to claims 1 to 3, characterised in that the AO generation block (805) comprises M inputs which are linked to the M least significant bits of the m-bits information relating to the position of the items on the page, and in response to the information on the inputs generates a bit pattern to the n-outputs which is used to address those memory elements from the pixel column memory (812, 813) which have to contain the shifted-out information at an address which is located one higher than the address location indicated by the (m - M) most significant bits of the Y position of the n-bits word which is being processed.

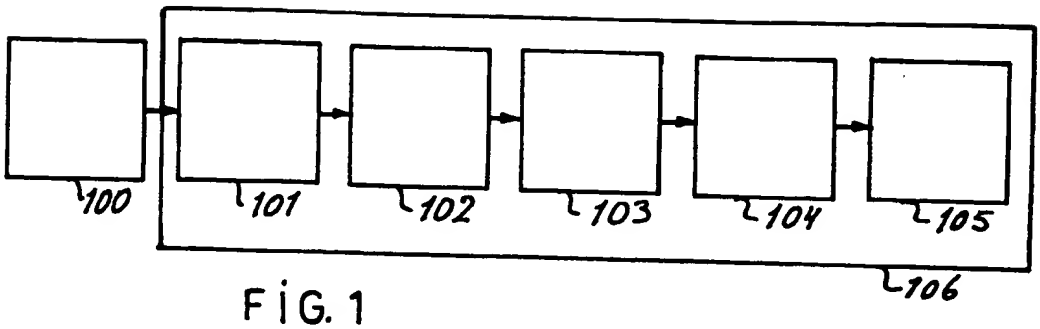

100
101
102
103
104
105
106

FIG. 1

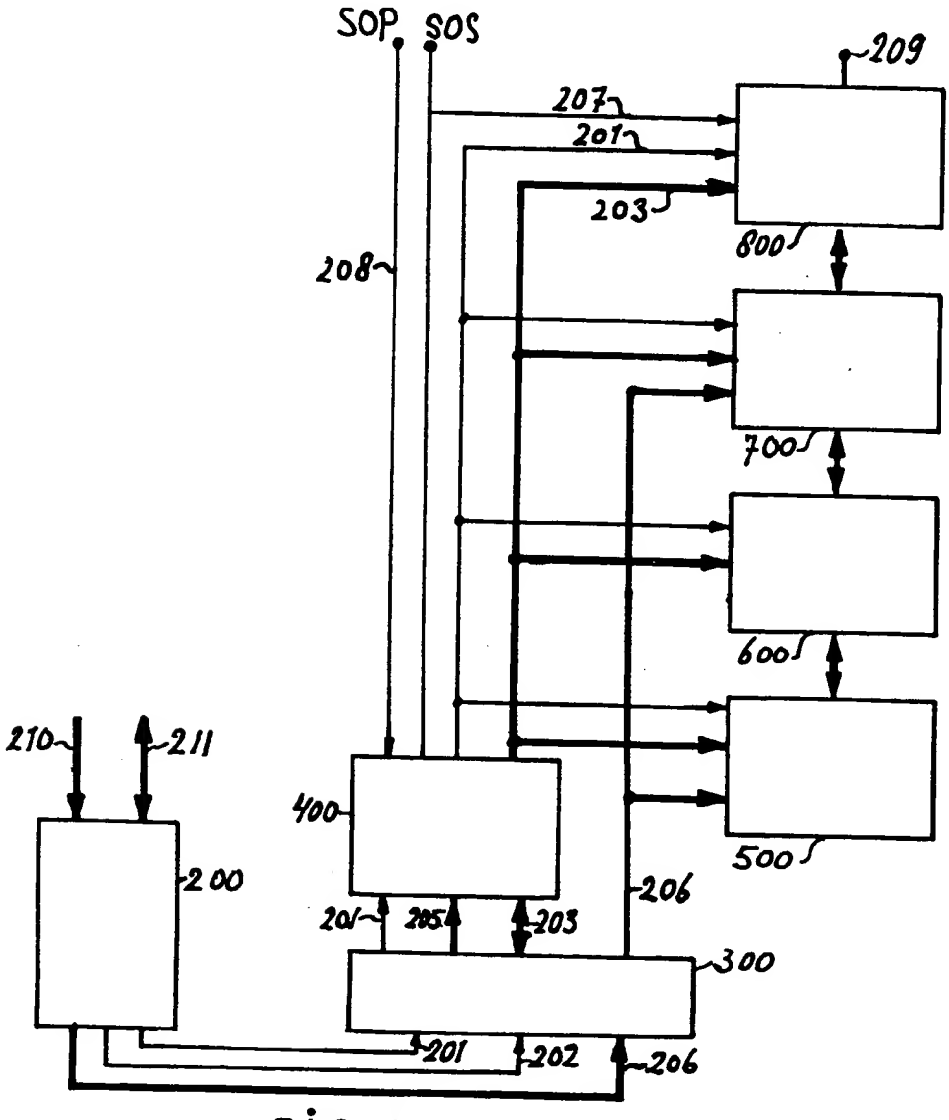

SOP
SOS
209
207
201
203
208
800
700
600
500
210
211
400
200
206
201
205
203
300
201
202
206

FIG. 2

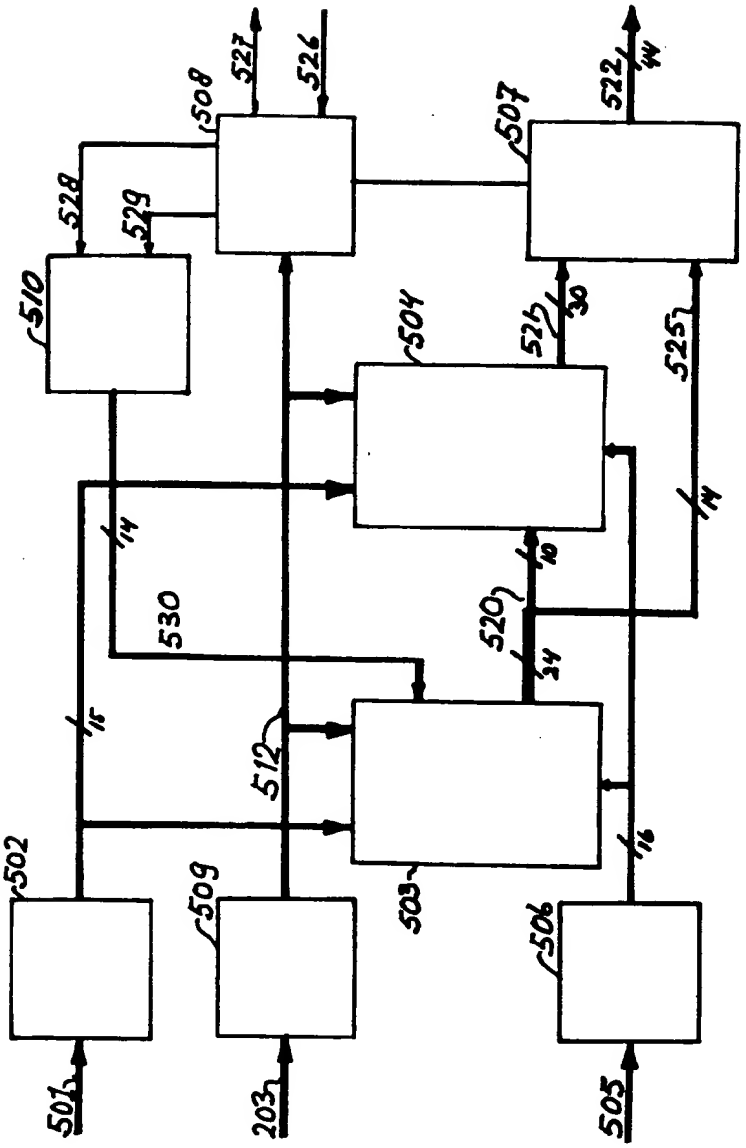

501
203
505
502
509
506
503
512
504
507
508
510
520
522
524
525
526
527
528
529
530
14
15
16
17
18
19
20
21
22

FIG. 3

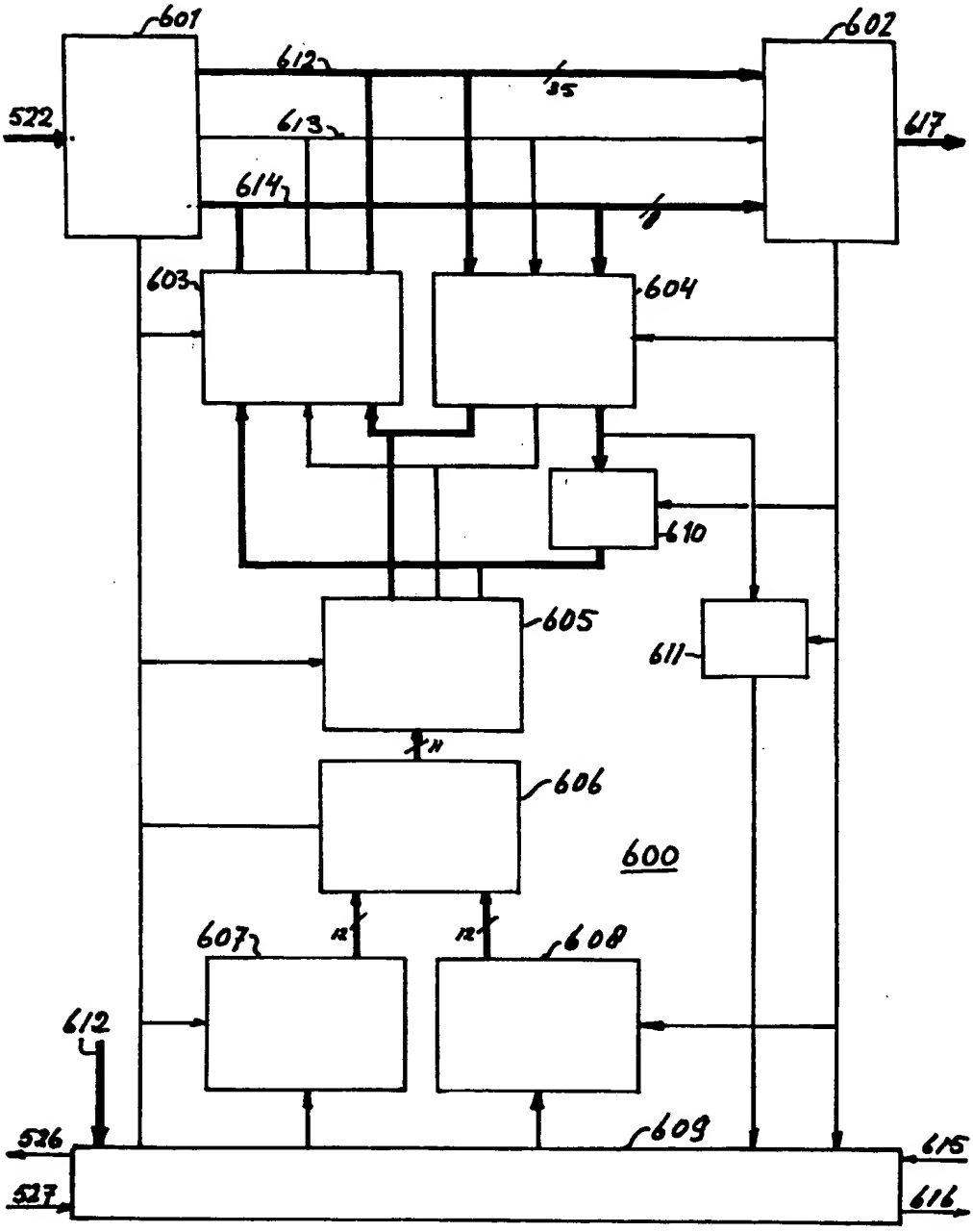

601
602
612
35
522
613
617
614
8
603
604
610
605
611
N
606
600
607
N
N
608
612
609
526
615
527
616

FIG. 4

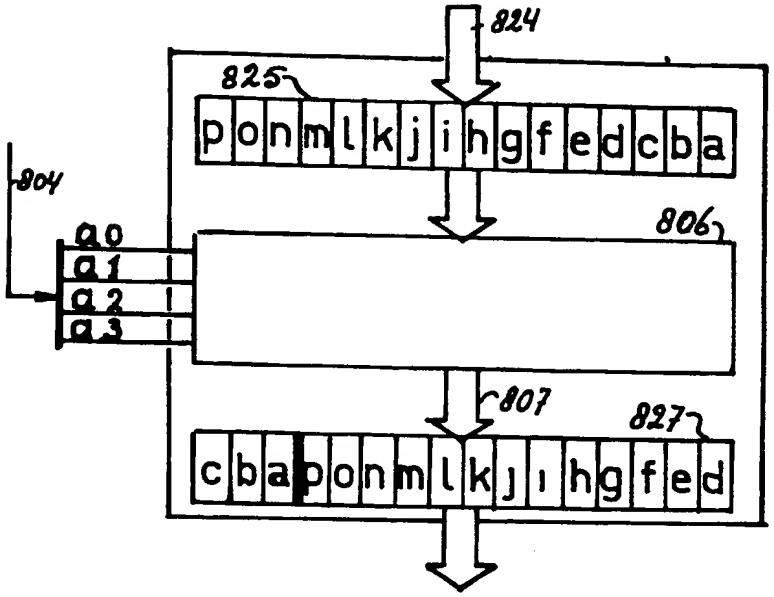

824
825
p o n m l k j i h g f e d c b a
804
806
a 0
a 1
a 2
a 3
807
827
c b a p o n m l k j i h g f e d

FIG. 8

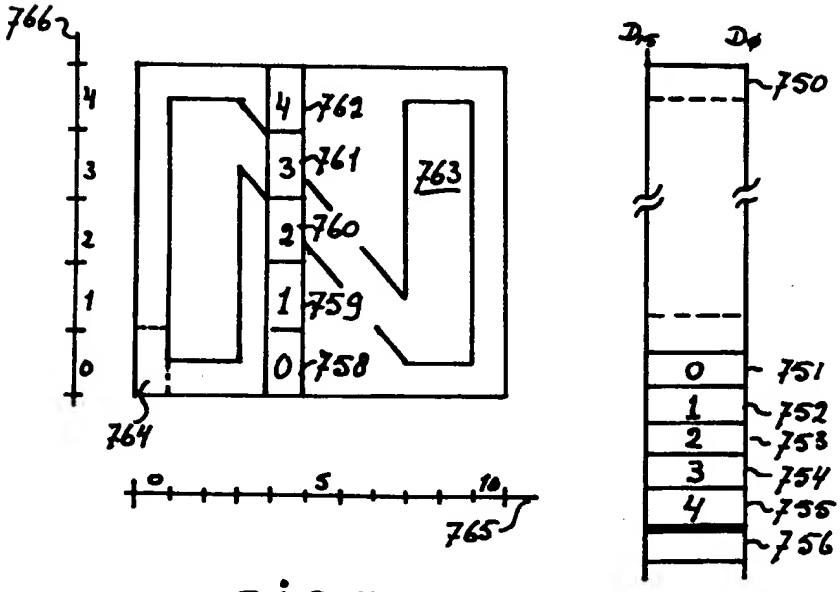

766
4
3
2
1
0
762
761
760
759
758
763
764
0
5
10
765
D5
D0
750
751
752
753
754
755
756

FIG. 5

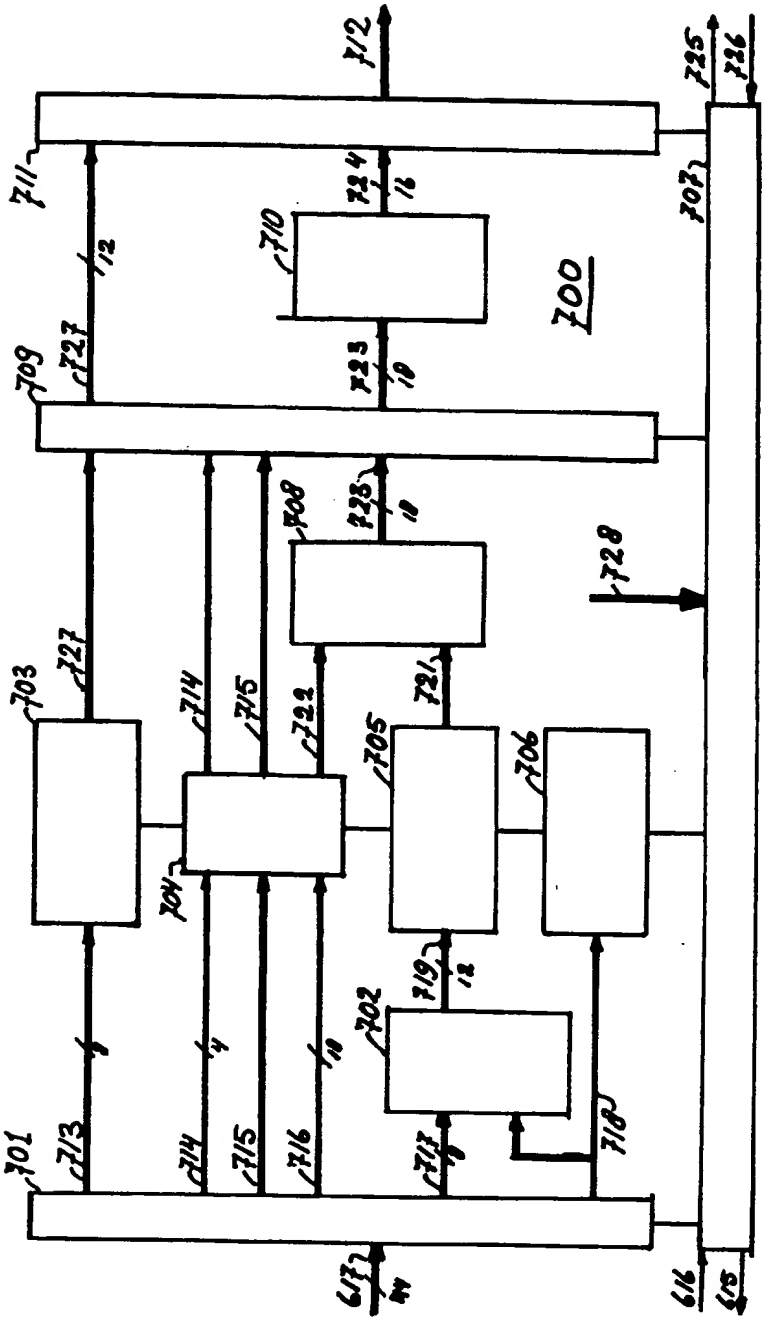

712
725
726
711
724
16
710
700
712
709
723
723
16
708
728
728
16
703
727
714
715
722
721
705
706
704
719
12
702
701
713
714
715
716
717
718
717
44
716
45

FIG. 6

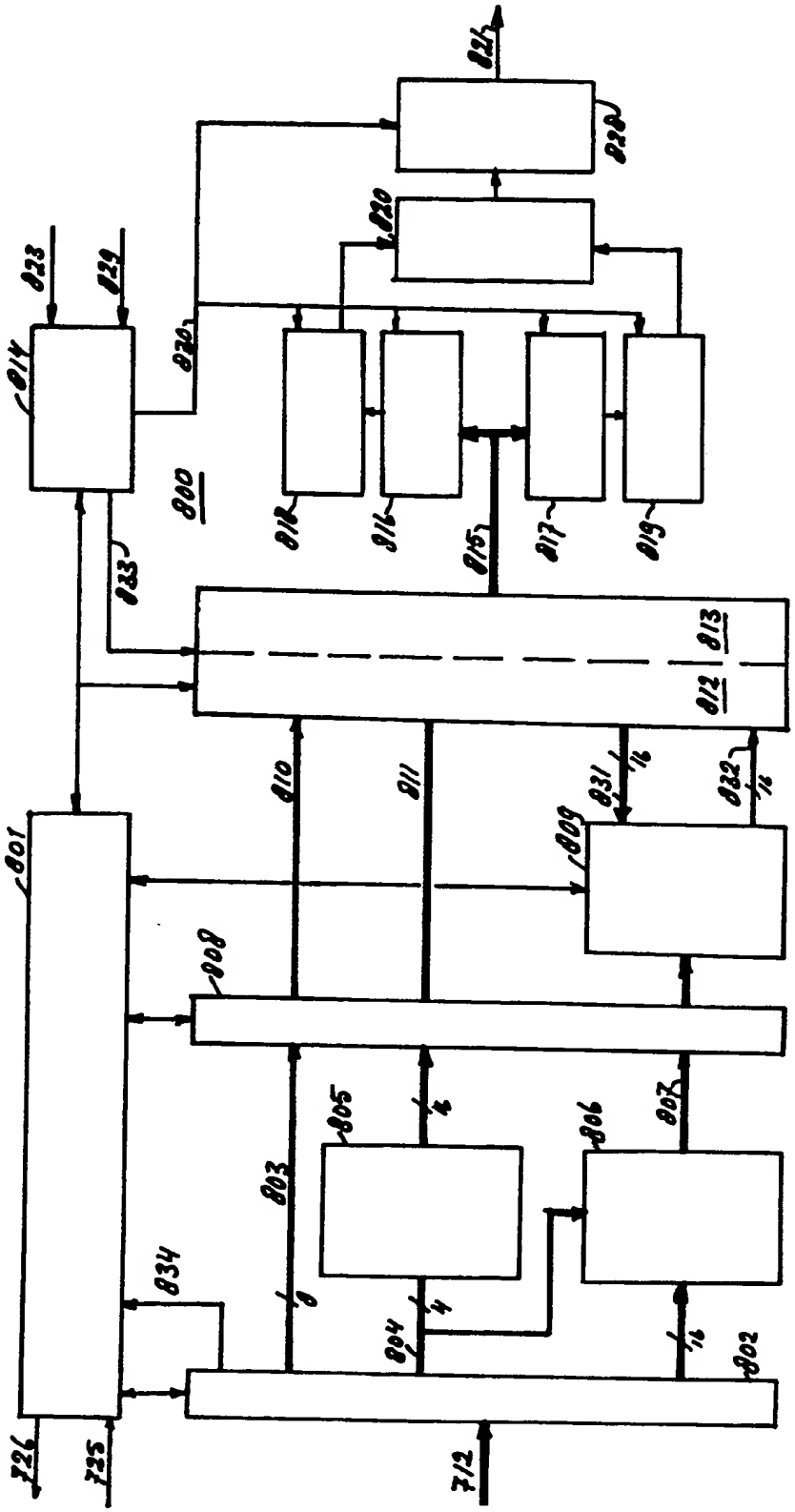

124
120
123
122
114
130
800
118
116
115
117
119
133
113
112
110
111
131
132
109
101
108
105
106
107
103
134
104
102
126
125
112

FIG. 7

European Patent Office

EUROPEAN SEARCH REPORT

0 180 258

EP 85 20 1527

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.4)
A,D	US-A-4 079 458 (RIDER) * Column 6, line 10 - column 8, line 44; figures 1-10 *	1	G 06 K 15/02 G 06 K 15/12
A	WO-A-8 303 487 (THE BOARD OF TRUSTEES OF THE LELAND STANFORD JUNIOR UNIVERSITY) * Pages 5-9 *	1	
A	GB-A-2 119 982 (XEROX CORP.) * Abstract; page 2, lines 71-77; page 4, lines 29-57; page 5, lines 21-44,75-102; figures 2-6 *	1	
A	US-A-4 153 950 (NOSOWICZ) * Column 2, line 12 - column 3, line 5; figure 1 *	1	TECHNICAL FIELDS SEARCHED (Int. Cl.4)
A	US-A-4 300 206 (BELLESON) * Abstract; figures 1-3 *	1	G 06 K

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	06-01-1986	PESCHEL W.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO Form 1503. 03.82

(19) Europäisches Patentamt
European Patent Office
Office uropéen des brevets

(11) Publication number: **0 180 258**
B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication of patent specification: **28.02.90**

(51) Int. Cl.[5]: **G 06 K 15/02**, G 06 K 15/12

(21) Application number: **85201527.0**

(22) Date of filing: **24.09.85**

(54) Decoder.

(30) Priority: **02.10.84 NL 8402998**

(43) Date of publication of application:
07.05.86 Bulletin 86/19

(45) Publication of the grant of the patent:
28.02.90 Bulletin 90/09

(84) Designated Contracting States:
DE FR GB IT NL

(56) References cited:
WO-A-83/03487
GB-A-2 119 982
US-A-4 079 458
US-A-4 153 950
US-A-4 300 206

(73) Proprietor: **Océ-Nederland B.V.**
St. Urbanusweg 43
NL-5914 CC Venlo (NL)

(72) Inventor: **Peters, Hendrikus**
Laan ten Boomen 9A
NL-5715 AA Lierop (NL)
Inventor: **Jongen, Alphons**
Bergerweg 26
NL-6063 BR Vlodrop (NL)
Inventor: **Tunnissen, Franciscus**
Frederik Hendrikstraat 57
NL-5923 BK Venlo (NL)

(74) Representative: **Hanneman, Henri W.A.M. et al**
Océ-Nederland B.V. Patents and Information
Postbus 101
NL-5900 MA Venlo (NL)

EP 0 180 258 B1

Courier Press, Leamington Spa, England.

Description

This invention relates to a decoder for converting information supplied in cod about a compiled page into image information which is fed in the form of a serial pixel bit flow to a raster output scanner, in which the raster output scanner is designed to form with the aid of a number of scanning lines an image on an image-forming medium, comprising a first memory for the storage of all items occurring on the compiled page together with the m-bits information relating to their position on that page representing a sorted table of items (STI), a second memory for the storage of the bit patterns of all items occurring in the first memory representing a pixel pattern data table (PPD), a third memory for the storage of the address information relating to the bit patterns in the second memory and for the storage of format information for items representing a table of pointers to pixel pattern data (TPPD), means for converting data taken from the sorted table of items, the pixel pattern data table and the table of pointers to pixel pattern data into n-bits words for each scanning line, and at least one n-bits wide pixel column memory.

Such a decoder, specifically suited for a xerographic laser printing system, is described in the United States Patent No. 4 079 458. In that patent, data relating to the characters to be printed on lines are sorted in such a way that all characters which start on one and the same scanning line are grouped together. The data of the characters which start on successive scanning lines are stored in an input memory, and the data relating to partially processed characters are stored in an active memory. Pixel pattern data for a number of characters are stored at separately accessible address locations in a font memory and these data are retrieved as a function of the data from the input memory and the activity memory during the successive scanning lines. The retrieved data are placed in an output memory and are then fed in the form of a serial pixel bit flow to a xerographic printer. To increase the processing speed, two output memories have been fitted. Whilst data are being read out of the first output memory, the second output memory can be filled. Nonetheless, no very fast processing speed can be achieved using the known decoder. The displacement of the word data in conformity with their relevant addresses and also the design of the output memories themselves are not conducive to fast decoding.

The object of the invention is therefore to provide a decoder in which these drawbacks are eliminated.

In the case of a decoder as described in the introduction, this object is achieved in that the decoder comprises:

rotation logic designed to shift around the n-bits words in a register over a number of register-places that is a function of the M l ast significant bits of the m-bits inf rmati n about th p sition of the items on the pag as stored in the first memory, whereby $2^M = n$ applies, an A0 generationblock generating a word-address in the pixel column memory which is double that of th m-M most significant bits of said position information whereby the information not shifted around (d—p) of the register of the rotation logic is stored in the pixel column memory at that address, and whereby the information of the register which is shifted around (a—c) is stored at the next higher word-address location in the pixel column memory, a first parallel-in-serial-out shift register which can be connected via a buffer to an odd address location of the pixel column memory, a second parallel-in-serial-out shift register which in relation to the first shift register, can be connected to a next-higher address location of the pixel column memory, a counter for generating successive addresses for reading out that pixel column memory and a combination circuit linked to the serial-out connections of the first and the second shift registers whereby both shift registers are clocked out simultaneously and serial data is combined in the combination circuit to form the serial pixel bit flow to the raster output scanner.

In the entire decoder parallel data processing is applied so that the conversion of information supplied in code into the required image information can take place especially rapidly. Moreover, it is possible to make do with little memory capacity.

Other features and advantages will be made clear in the following description, in which reference is made to the attached drawings, of which:

Fig. 1 is a schematic representation of the location of the decoder in an image processing station,

Fig. 2 is a schematic representation of the decoder,

Fig. 3 is a block diagram of the item processor of the decoder,

Fig. 4 is a block diagram of the item-slice processor of the decoder,

Fig. 5 represents the structure of a pixel pattern of an item,

Fig. 6 is a block diagram of the pattern data generator of the decoder,

Fig. 7 is a block diagram of the column data generator of the decoder, and

Fig. 8 is a block diagram of a barrel-shifter.

On an input station 100 Fig. 1 (word processor) a page is typed in and sent to the printer 106. In the printer 106 the incoming page is converted with the aid of translator 101 into a code suitable for the printer, the UIL (universal input language). This information in UIL code is fed to the formattor 102 in which the formatted page is compiled. This contains each character (or font) with its X and Y positions on the page and its font code. The formatted-pag inf rmation is now arranged in the converter 103 to make what is kn wn as a converted page, suitable for th dec der 104. The decoder 104 translates the arranged inf rmati n into a vide signal which is

used to modulate the laser beam of th laser printer 105. With the aid of this laser beam an electrophotographic image is written in accordance with the known method and th laser printer 105 prints out this image on plain paper. The printer which is used with the decoder according to the invention is designated to process a page in the landscape form as well as in the portrait form, but in order to give a clear description of the printer it has been assumed that a page of text in the portrait form is supplied from a word processor. The laser printer reproduces this information rotated through 90 degrees. The terms X and Y direction relate to the information originally supplied, in other words the X direction is the direction parallel to the lines and the Y direction or SCAN direction is the direction at right-angles to the lines.

Via the converter 103, the decoder 104 receives three different tables, known as the decoder tables. The first table is called the "sorted table of items" (STI) and contains all items which occur on a page in their sorted X sequence, whereby an item is understood to mean a letter, number or logo. All items which start with a same X value are grouped and a set of items with the same X value is referred to as a column. All columns are closed with an "end of column" (EOC) bit. This EOC bit is also present where a column remains empty if not a single item starts with the relevant X value. If printing has to be done at a resolution of 12 dots/mm, about 2520 columns are needed for one A4 size page. This means that the STI table in fact consists of 2520 sub-tables of varying lengths. If a lot of items start with one specific X value, such a sub-table will be long (for example, 600 locations), whilst a sub-table for an empty column will consist of one location.

Each element of the STI table contains an item number (this is a unique code for each character or logo) and the Y starting value for that item on the page to be printed. Another table of the decoder tables is the "Table of Pointers to Pixel Pattern Data" (TPPD), which is a vector table. The vector is an address in the pixel pattern memory. In this table the relevant information on the height and width of the item is also stored for each item number. The height of an item is indicated in units of 16 pixels, the width in units of 1 pixel. The last table of the decoder tables is the "Pixel Pattern Data" table (PPD) which contains the pixel pattern of all items occurring on the page and which can best be described as a sort of loadable character memory.

The items are built up of pixel words of 16 pixels. The collection of all 16-bits pixel words in one item, sorted according to their ascending Y position and with one value for X, is referred to as an item slice.

Once the decoder has been loaded with the above-mentioned tables, the decoding process can start. One column is r ad out of th STI table each time and ach item occurring in this is expanded with th aid of the TPPD table and the pixels are supplied to a line buffer. If an item in a preceding scan has still not been completely processed, the remaining information from it is plac d in a tally memory. Each el m nt in this tally memory contains the inf rmati n on the height of th item, the width that remains to be processed, the Y position and the address in the PPD table where the relevant pixel pattern of the item being processed is located.

Each element from the tally memory therefore comprises two types of information, viz. information which makes it possible to address the memory in which the PPD table is located, and information relating to the Y position which indicates the starting point from which the relevant pixels for this item should be placed on paper. During each scan (Y cross-section of the page) all elements of the tally memory are processed and, if an item has still not been fully processed, it is put back in the tally memory. If all elements of a scan have been processed, new elements of a following column are processed and, here again, the still not fully processed items are put back in the tally memory. All pixels of a scan are read into a first line buffer and are read out in response to a command from the laser printer whilst a second buffer line is being filled with the information for the following scan.

Fig. 2 gives a schematic representation of the decoder. The actual decoder part consists of the item processor 500, the item-slice processor 600, the pattern data generator 700 and the column data generator 800. For controlling purposes a monitor 200, page-control 400 and a bus-interface 300 have also been added.

The information from a page is passed via bis 210 to monitor 200, whilst the decoder dialogue between an input station and the monitor 200 is transmitted via bus 211. The system-clock line 201 is linked to the other modules via bus-interface 300.

Controlling signals, such as the "decode-a-page" signal, are linked to the various modules via bus-interface 300 through bus 203. The information on the contents of a page is supplied to the modules via bus 206 and bus-interface 300.

The function of page-control 400 is page synchronisation: synchronising the laser printer and the decoder with a "decode-a-page" command.

The item processor 500 comprises the memories for the STI and TPPD tables. They are loaded via bus 206. The item processor 500 also supplies items in response to requests from the item-slice processor 600.

The item-slice processor 600 receives items from the item processor 500 and supplies item-slices in response to requests from the pattern data generator 700. Items which consist of several slices are temporarily stored in a tally memory in the item-slice processor 600.

The pattern data generator 700 receives item-slices from th item-slice processor 600. Via bus 206 the 16-bit pix l patterns are retrieved across their full height from the PPD memory and are supplied, together with their relevant Y positions, to the column data generator 800 upon request.

Th column data g nerator 800 receives th pix l patterns and the Y position of an item via pattern data generator 700. For the laser printer a pixel bit flow 209 is generated, which supplies the pixel patterns serially according to their Y sequence. Pixel patterns with an identical Y position (overlay) are combined. A "start-of-page" signal (SOP) is fed from the laser printer via line 208 to page-control 400. The "start-of-scan" signal (SOS) is fed via line 207 to page-control 400 and to the column data generator.

Item processor

The operation of the item processor will be explained by reference to Fig. 3. Via buffer 502 the addresses of the items occurring on a page which are supplied on bus 501 are fed to STI & STI Control block 503 as well as to TPPD & TPPD Control block 504. Via buffer 506 the corresponding data supplied on bus 505 are likewise fed to block 503 and block 504. Buses 501 and 505 form part of bus 206 as indicated in Fig. 2. Via buffer 509 the internal control bus 512 is linked to bus 203.

Blocks 503 and 504 both comprise a memory and a control section. In a period when no "decode" signal is present on bus 512, the memories in blocks 503 and 504 are filled with information about a compiled page. The STI table is read into the memory of block 503 and the TPPD table into the memory of block 504. The STI table is a 24-bits wide table which is subdivided into 12 bits which indicate the item's Y position on the page, 10 bits for the item number, one "end of column" bit (EOC) and one reverse bit. If a "decode" signal is supplied to bus 512, then the information relating to the items occurring on a page will become available item by item from the STI memory via buses 520 and 525. The 10 bits which represent the item number are supplied via bus 520 to the TPPD & TPPD control block 504. They constitute the address for the TPPD table which is present in the memory of block 504. The remaining 14 bits of the STI table are fed via bus 525 to an output register 507.

The TPPD table which is present in block 504 is 30 bits wide. Of these bits, 18 bits indicate the PPD base address whilst, of the remaining 12 bits, 8 bits represent the width of the item and 4 bits represent the height of the item.

During the decode state these data are likewise supplied from the TPPD table via bus 521 to the output register 507. Bus 522, which connects the output register 507 with the item-slice processor 600 and which is 44 bits wide, thus holds all information relating to one item, viz. the Y position on the page (12 bits), the PPD base address (18 bits), the width (8 bits) and the height (4 bits), the EOC bit and one reserve bit.

The item processor 500 only supplies data to the item-slice processor 600 at the latter's request. For this purpose a data-request signal 526 is fed to output c ntrol 508. The item process r transmits a data-availabl signal 527 t the item-slice process r if the request d data are pr sent in output register 507.

A counter 510 is linked t the STI & STI C ntrol block 503. The counter readings are used to read out the STI table sequentially, whilst the counter reading indicates the address in the table. When a decode-a-page signal is sent via the item-slice processor 600 to output control 508, the latter generates a counter-reset signal which is supplied to the counter via line 528. The read-out of the STI table starts at address 0. At every data-request signal 526, the counter reading is increased by one via line 529.

Item-slice processor

Fig. 4 is a schematic representation of the item-slice processor (ISP). During the decoding process the item-slice-processor control 609 (ISP control) generates a data-request signal 526 for the item processor 500. If this item processor is ready for data transmission, it gives a data-available signal 527 to the ISP control 609. Then one column is retrieved from the item processor 500 through bus 522 and via input register 601 and this column is passed on to the pattern data generator 700 via output register 602 and bus 617. One column contains all items having the same value of X which occur on a page.

The 44 bits wide bus 522 is divided up in the ISP into a 35 bits wide bus 612, which transports the PPD base address, the height of the item and its Y position, a line 613 through which an EOC bit can be transported, and an 8 bits wide bus 614 which transports information relating to the width − 1 of an item.

If an item is composed of more than one slice (which will usually be the case), the data of that item are stored via register 604 into the tally memory 605. This tally memory is a 2K × 48 bit memory. The bit information relating to the width of the item is reduced by one with the aid of a subtraction circuit 610 and is also stored in the tally memory 605. The portion of an item that is left over after one or more slices have already been removed is referred to as the item remainder. If a first column has been sent in this way via bus 617 to the pattern data generator 700, which is indicated by an EOC bit, then a start can be made on the second column.

For this purpose the item remainder of the first item is supplied from the tally memory 605 via buffer 603 and output register 602 through bus 617 to the pattern data generator 700. The W = 0 detector 611 serves to ascertain whether the width − 1 information relating to the item or item remainder being processed has decreased to zero. If this is not the case, the bit information relating to the width of the item is again reduced by one with the aid of the subtraction circuit 610. This new item remainder is stored in the tally memory 605 again. If the width − 1 information has decreased to zero, then the item has been fully processed and is therefore no longer fed via register 604 to the tally memory.

After all items st r d during the previous column have been processed in the tally memory 605, the new items which bel ng to this column

are retrieved from the item processor and dealt with as described above.

To enable effective use to be mad of the tally memory 605, two counters 607 and 608 as well as an address selection block 606 have been added.

During writing to the tally memory 605, this memory is addressed with the aid of write-counter 608 via the address selection block 606. At the start of the reading-in of the first column from the item processor 500, write-counter 608 is reset to zero. The first item remainder fed to the tally memory 605 is thus written to address zero.

Subsequently, the counter reading of write-counter 608 is increased by one so that the second item remainder from the first column is written to address one.

All item remainders from the first column are then written successively by ascending address into the tally memory 605. The last item remainder is placed at address N. After that, an EOC bit which originates from the item processor is stored at address N + 1.

When the second column is being processed, read-counter 607 and write-counter 608 are first reset to zero. The item remainders in the tally memory 605 are read out in conformity with the address indicated by counter 607. In other words, the item remainder at address zero is read out first. If a new item remainder has again been left over from this item remainder, such remainder will again be written to address zero, as indicated by write-counter 608.

The read-counter 607 is increased by one and the following item remainder is read out. The write-counter 608 is only increased if a remainder has again been left over from the item or item remainder which is being processed. In this way the tally memory 605 is always filled from address zero onwards. The read-out of the tally memory 605 continues until the EOC bit is read. After that, new items forming part of the columns being processed are retrieved from the item processor 500, are processed, and any item remainders are successively stored in the tally memory 605 until an EOC bit is received again. Item-slice-processor control 609 consists of a Field Programmable Logic Array (FPLA) and a register, which together constitute a "State machine". The FPLA determines the new state on the basis of the information supplied to the inputs and the state which is being processed. The information emerging at the output simultaneously determines the new state being processed.

The operation of the various blocks of the item-slice processor is controlled by the item-slice-processor control 609. The control communication between the item-slice processor and the other parts of the decoder takes place via bus 612. The data-available signal 527 from the item processor and the data-request signal 615 from the pattern data g nerator are linked with item-slice-processor control 609, as are th data-request signal 526 and th data-availabl signal 616.

The pattern data generator

The principal task of the pattern data generator is to translate the item-slice information supplied by the item-slice process r into pixel patterns. The item-slice information comprises the data relating to the height of the item, the Y position on the page, the base address of the PPD memory at which the pixel information for the relevant item has been stored, and the item-progress. The item-progress is understood to mean the item remainder width − 1 information. After being processed in the pattern data generator, the 16-bits words which represent the pixels in a slice are supplied, together with their relevant Y positions, to the column data generator 800 in response to its request.

Fig. 5 represents how the pixel patterns of an item are arranged in the PPD memory. The example used is a letter N (763), which has been divided into a matrix of 11 × 5 sectors. The item-progress has been plotted along horizontal axis 765, whilst the height minus 1 in words (= H) has been plotted along the vertical axis 766. The starting dot of the letter is sector 764. When defining the X and Y position of this item on the page, the position is related to this sector 764. The base address of this item is the address of memory location 750 where the pixel pattern of sector 764 is to be found.

An item-slice with 5 consecutive 16-bits words is indicated by the numbers 758 to 762 inclusive. The item-progress indicated on axis 765 is four for this slice. The item-slice word 758 is placed in the memory at location 751. The word 759 is at memory location 752 and word 762 is at memory location 755.

The address of memory location 751 corresponds to the base address 750 plus the product of the height minus 1 of the item + 1 (= H + 1) and the item-progress. If the base address 750 of the PPD memory were 1000 (hex$), then the address of word 758 and hence of memory location 751 in the PPD memory would be 1015 (hex$).

In Fig. 6 bus 617 is the connection between the output register 602 of the item-slice processor 600 and the input register 701 of the pattern data generator 700. This 44-bits-wide bus 617 is divided up in such way that the offset calculation block 702 receives the bit pattern forming part of the item-progress (Ip) (8 bits) via bus 717 and the bit pattern forming part of the item height (H + 1) (4 bits) via bus 718. In block 702 the product of the item-progress (Ip) and the height (H + 1) is calculated (Ip × (H + 1)). The offset signal thus calculated (12 bits) is sent through bus 719 to the offset address counter 705. In summing circuit 708 the offset address is added to the base address (18 bits), which is supplied via bus 716, register 704 and bus 722. Bus 723 is thus given the address of th first word of th relevant item-slice in th PPD m mory 710.

Subsequently, the height of the item-slice being processed is counted down to zero in height-count r 706. For each subtraction step of the

height-counter 706 the contents of th offset address c unter 705 are increased by one and this increased value i in turn added to th base address in summing circuit 708. In this way the consecutive addresses which form part of a specific item-slice are selected in the PPD memory.

The 12-bits-wide information supplied through bus 617 relating to the Y position of an item on a page is divided up into the 8 most significant bits, which are fed via bus 713 to the Y-destination counter 703, and the 4 least significant bits, which are fed via bus 714 to register 704. These four least significant bits of the Y position represent the shift information. Via bus 714 and pipeline register 709, this shift information is supplied via output register 711 to bus 712.

For each subtraction step of the height-counter 706, the contents of the Y-destination counter 703 are also increased by one. This new Y-position value is sent through bus 727 to output bus 712.

The pixel pattern which belongs to the address supplied on bus 723 is fed to the column data generator 800 via the output register 711 through bus 712.

The pattern-data-generator-control block 707 ensures the correct operation of the generator with the aid of control signals which are sent to the various counters and registers. Via bus 728 this control block 707 communicates with other control blocks in the decoder (clock signals, decode signal, etc.). A data-available line 725 and a data-request line 726 have been installed between the column data generator 800 and the pattern data generator 700.

The column data generator

The task of the column data generator 800 (Fig. 7) is to convert the (parallel) information that has been supplied into a serial pixel bit flow for the laser printer. Via bus 712 the pipeline register 802 of the pattern data generator 700 receives the pixel bit pattern of the items (16-bits words) and also the relevant Y positions in respect of one specific X position. At this point the sequence is still arbitrary. These pixel bit patterns have to be sorted according to their ascending Y position, whilst pixel bit patterns with an identical Y position (overlays) have to be combined.

Since rapid signal processing is essential, the pixel column memories 812 and 813 are designed as separate 16-bits-wide and 512-words-deep memories. Whilst one pixel column memory is being read out, the other pixel column memory is being filled with 16-bits words. To make it possible to address the 16-bits words in the pixel column, use is made of the 8 most significant bits of the Y position. So that it is still possible to indicate the position of an item on a page with an accuracy of 1 pixel whilst not having to abandon the simultaneous processing of 16-bits words, the four l ast significant bits, prior to their b ing stored in one of th pixel column memories 812 and 813, are used to nsure that the bits in the separate 16-bits w rds are displaced in such a way that the starting dot of an item (the word border) corresponds to the required Y position.

This displacement is effectuated in a rotation logic block 806 with th aid f a barrel-shift r. The operation of this barrel-shifter 806 will be explained by reference to Fig. 8.

Via data-bus 824 the 16-bits-wide words, which represent the pixel pattern of the item being processed are fed from the PPD memory to the input register 825 of the barrel-shifter 806.

Of the 12-bits information relating to the Y position of that pattern on the page, the 4 least significant bits are separated off and are also fed to barrel-shifter 806 via address bus 804. Depending on the address information on bus 804, the information which is present in input register 825 is shifted and stored in output register 827.

Input register 825 and output register 827 form part of pipeline register 802 and pipeline register 808 respectively.

In input register 825 the letters a to p inclusive represent the contents of the register locations which are coupled to the respective data lines D0 and D15 inclusive. If it is assumed that at address bus 804 (via address lines A0 to A3) the least significant bits of the Y position correspond to, say, binary 3 (= 0011), then the information from input register 825, shifted by three places, will be fed into the output register 827. The information that has been shifted around (a, b and c) will again be stored at the front in output register 827.

To enable high-speed processing, the barrel-shifter 806 is designed in such a way that shifting is effected in parallel, on all 16 bits at the same time.

The pixel column memories 812 and 813 are each composed of 16 memories of 512 × 1 bit which are used in parallel for writing-in of 16-bits words. These memories are addressed by using the 8 most significant bits of the Y position (the Y-word address), which are fed to the addresses A1—A8 of the 16 memories via pipeline register 802, bus 803 and pipeline register 808 through bus 810. The A0 address line of each memory IC is linked via bus 811 to A0 generation block 805.

The A0 generation block 805 has an input bus 804 through which the 4 least significant bits of the Y position are supplied. Depending on these inputs signals, one or more of the 16 outputs, each of which is linked to the A0 address of one memory IC of memory 812 or 813, may become a logical 1 or 0. For instance, if a signal supplied to the input corresponds to binary 3 (= 0011), then three output lines which are linked to the A0 address lines of three memory ICs from pixel column memory 812 or 813 will become a logical 1, which means that these memories will be addressed at the next-higher word address compared to the other 13 memories. This makes it possible for the information shifted around in rotation logic block 806 (f r example, bits a, b and c in Fig. 8) to be written to the next-higher w rd address in the pixel column memory 812 or 813. The overlay-l gic block 809 c mprises a register to which the data are supplied from the output

register f barrel-shifter 806, which is situated in pipeline register 808. The pixel column memories 812 and 813 ar addressed using the word address p rtaining to thos data, whilst the data already present at that address are also fed via bus 831 to overlay-logic block 809. These data are combined via an OR relation with the data resident in the overlay register, and the result of this is written back again to the selected address.

If one pixel column memory has been completely filled with the information from a scanning line, this memory is switched to the read-out state, whilst the other memory is changed from the read-out state into the write-state. Output control 814 gives the command for this to the pixel column memories via bus 833. During the time that one memory is being filled, the other memory is read out.

Output bus 815 of the pixel column memories 812 and 813 is linked to two registers 816 and 817, each of which is one word wide and each of which is connected through to a 16-bits shift register, 818 and 819 respectively. Output control 814 supplies the output addresses A0 to A8 inclusive to the pixel column memory 812 or 813 with the aid of a counter and via bus 833. During this read-out all A0 address lines of the 16 memory ICs are interlinked. The 16-bits data word at address location 1 of the pixel column memory is transmitted in parallel via bus 815 to register 817. Then, address location 1 is cleared (set to zero) and the address is increased by one, with the result that address location 2 is selected. The data at address location 2 are supplied to register 816 via output bus 815. Address location 2 is now cleared. In this same manner the shifted around bits of a word, which had been stored at an odd address location are supplied each time to shift register 819, whilst the non-shifted around bits of the following word, which had been stored at an even address location, are supplied to shift register 818.

The laser printer generates an internal burst signal which corresponds to the required write-frequency (the pixel-bit rate) of the pixel bit flow to be fed to the printer. This burst signal is transmitted via line 829 to output control 814 and is then distributed to shift registers 818 and 819 and to flip-flop 828. By means of this burst signal both the shift registers 818 and 819 are serially clocked out at the same time and the data from these registers are combined in combination circuit 820.

The output of combination circuit 820 is transmitted to flip-flop 828 and the burst signal ensures that the pixel bit flow to the laser printer via line 821 is synchronised when it leaves the column data generator 800 via line 821.

The control of the input logic of the column data generator is handled by input control 801. Data-request line 726 and data-availabl line 725 are also connected t input control 801. Th EOC bit is also fed t input contr l 801 via lin 834.

Claims

1. Decoder for converting information supplied in code about a compiled page into information which is fed in th form of a serial pixel bit flow to a raster output scanner, in which the raster output scanner is designed to form with the aid of a number of scanning lines an image on an image-forming medium, comprising a first memory for the storage of all items occurring on the compiled page together with the m-bits information relating to their position on that page representing a sorted table of items, a second memory for the storage of the bit patterns of all items occurring in the first memory representing a pixel pattern data table, a third memory for the storage of the address information relating to the bit patterns in the second memory and for the storage of format information for items representing a table of pointers to pixel pattern data, means for converting data taken from the sorted table of items, the pixel pattern data table and the table of pointers to pixel pattern data into n-bits words for each scanning line, and at least one n-bits wide pixel column memory, characterised in that the decoder comprises rotation logic (806), designed to shift around the n-bits words in a register (827) over a number of registerplaces that is a function of the M least significant bits of the m-bits information about the position of the items on the page as stored in the first memory (503), whereby $2^M = n$ applies, an A0 generationblock (805) generating a word-address in the pixel column memory (812, 813) which is double that of the m-M most significant bits of said position information whereby the information not shifted around (d—p) of the register (827) of the rotation logic (806) is stored in the pixel column memory (812, 813) at that address, and whereby the information of the register (827) which is shifted around (a—c) is stored at the next higher word-address location in the pixel column memory, a first parallel-in-serial-out shift register (818) which can be connected via a buffer (816) to an odd address location of the pixel column memory (812, 813), a second parallel-in-serial-out shift register (819) which in relation to the first shift register (818), can be connected to a next-higher address location of the pixel column memory (812, 813), a counter for generating successive addresses for reading out that pixel column memory (812, 813) and a combination circuit (820) linked to the serial-out connections of the first (818) and the second (819) shift registers whereby both shift registers (818, 819) are clocked out simultaneously and serial data is combined in the combination circuit (820) to form the serial pixel bit flow to the raster output scanner.

2. Decoder according to claim 1, characterised in that the rotation logic (806) comprises a barrel-shifter which is connected t an n-bits input register (825), an n-bits output regist r (827), and a circuit which, as a function f M control signals, can shift the data pres nt in th input regist r (825) around and place them in th output register (827).

3. Dec der according to claims 1 or 2, characterised in that the pixel column memory (812, 813) is composed of n memory elem nts of which from each memory element one address lin is linked to one of the n outputs of the A0 generation block (805) and of which the remaining address lines are linked together.

4. Decoder according to claims 1 to 3, characterised in that the A0 generation block (805) comprises M inputs which are linked to the M least significant bits of the m-bits information relating to the position of the items on the page, and in response to the information on the inputs generates a bit pattern to the n-outputs which is used to address those memory elements from the pixel column memory (812, 813) which have to contain the shifted-around information at an address which is located one higher than the address location indicated by the (m-M) most significant bits of the Y-position of the n-bits word which is being processed.

Patentansprüche

1. Dekoder zum Umwandlen von in kodierter Form zugeführter Information über eine kompilierte Seite in Information, die in Form eines seriellen Pixel-Bitstromes in einen Raster-Abtaster als Ausgabegerät eingegeben wird, wobei der Raster-Abtaster so konstruiert ist, daß er mit Hilfe einer Anzahl von Abtastzeilen ein Bild auf einem Bilderzeugungsmedium erzeugt, mit einem ersten Speicher, der zur Speicherung aller auf der kompilierten Seite auftretenden Objekte zusammen mit der m-bit-Information bezüglich ihrer Position auf dieser Seite dient und eine sortierte Objekttabelle darstellt, einem zweiten Speicher, der zur Speicherung der Bitmuster sämtlicher im ersten Speicher auftretender Objekte dient und eine Pixelmusterdaten-Tabelle darstellt, einem dritten Speicher, der zur Speicherung von Adresseninformationen bezüglich der Bitmuster in dem zweiten Speicher und zur Speicherung von Formatinformationen für Objekte dient und eine Tabelle von Zeigern für Pixelmusterdaten darstellt, Mitteln zum Umwandeln von aus der sortierten Objekttabelle, der Pixelmusterdaten-Tabelle und der Tabelle der Zeiger für Pixelmusterdaten entnommenen Daten in n Bit-Wörter für jede Abtastzeile und mit wenigstens einem n-bit größen Pixelspalten-Speicher, dadurch gekennzeichnet, daß der Dekoder aufweist: eine Rotationslogik (806) zum zyklischen Verschieben der n-bit-Wörter in einem Register (827) über eine Anzahl von Registerplätzen, die eine Funktion der M geringstsignifikanten Bits der in dem ersten Speicher (503) gespeicherten m-bit-Information über die Position des Objektes auf der Seite ist, wobei die Beziehung $2^M = n$ gilt, einen A0-Erzeugungsblock (805), der eine Wortadresse in dem Pixelspalt n-Speicher (812, 813) rzeugt, die das zweifach der m-M h chstsignifikanten Bits der Positi nsinformation ist, wob i die Information (d—p), die in dem Register (827) der Rotationslogik (806) nicht zum Anfang zurückgesch ben wurde, in d m Pixelspalten-Speicher (812, 813) unter dieser Adresse gespeichert wird und wobei die Information (a—c), die zum Anfang des Registers (827) zurückgeschoben wurde, in dem Pixelspalten-Speicher an einem Platz mit der nächsthöheren Wortadresse gespeichert wird, ein erstes Schieberegister (818) mit parallelem Eingang und seriellem Ausgang, das über einen Puffer (816) mit Platz des Pixelspalten-Speichers (812, 813) mit ungerader Adresse verbindbar ist, ein zweites Schieberegister (819) mit parallelem Eingang und seriellem Ausgang, das in bezug auf das erste Schieberegister (818) mit einem Platz mit der nächsthöheren Adresse in dem Pixelspalten-Speicher (812, 813) verbindbar ist, einen Zähler zum Erzeugen aufeinanderfolgender Adressen zum Lesen des Inhalts des Pixelspalten-Speichers (812, 813) und eine Kombinationsschaltung (820), die mit den seriellen Ausgangsanschlüssen des ersten (818) und zweiten (819) Schieberegisters verbunden ist, wobei beide Schieberegister (818, 819) gleichzeitig ausgetaktet werden und die seriellen Daten in der Kombinationsschaltung (820) kombiniert werden, um den seriellen Pixel-Bitstrom zu dem Raster-Abtaster zu bilden.

2. Dekoder nach Anspruch 1, dadurch gekennzeichnet, daß die Rotationslogik (806) einen Rotationsschieber aufweist, der mit einem n-bit-Eingangsregister (825), einem n-bit-Ausgangsregister (827) und einer Schaltung verbunden ist, die als Funktion von M Steuersignalen in der Lage ist, die in dem Eingangsregister (825) enthaltenen Daten zyklisch zu verschieben und in dem Ausgangsregister (827) abzulegen.

3. Dekoder nach Anspruch 1 oder 2, dadurch gekennzeichnet, daß der Pixelspalten-Speicher (812, 813) aus n Speicherelementen besteht, wobei von jedem Speicherelement eine Adressenleitung mit einem der n Ausgänge des A0-Erzeugungsblocks (805) verbunden ist, während die übrigen Adressenleitungen zusammengeschaltet sind.

4. Dekoder nach den Ansprüchen 1 bis 3, dadurch gekennzeichnet, daß der A0-Erzeugungsblock (805) M Eingänge aufweist, die mit den M geringstsignifikanten Bits der m-bit-Information über die Position der Objekte auf der Seite verbunden sind, und in Abhängigkeit von der Information an den Eingängen ein Bitmuster an den n-Ausgängen erzeugt, das zur Adressierung derjenigen Speicherelemente aus dem Pixelspalten-Speicher (812, 813) dient, die die zurückgeschobene Information unter einer Adresse enthalten sollen, die um eins höher ist als der Adressenplatz, der durch die (m-M) höchstsignifikanten Bits der Y-Position des verarbeiteten n-bit-Wortes angegeben wird.

Revendications

1. Déc deur destin à conv rtir des informations fournies en cod , concernant une page compilée, en des informations qui sont fournies sous la forme d'un débit binair d'élém nts

d'image en série à un analyseur de sortie à tram , dans lequel l'analyseur de sortie à trame est réalisé de manière à former, à l'aide d'un certain nombre d lignes de balayage, un image sur un support de formation d'image, comportant une première mémoire pour la mémorisation de tous les articles apparaissant sur la page compilée avec les m bits d'information concernant leur position sur cette page, représentant une table classée d'articles, une seconde mémoire pour la mémorisation des configurations binaires de tous les articles apparaissant dans la première mémoire, représentant une table de données de configurations d'éléments d'image, une troisième mémoire pour la mémorisation des informations d'adresse concernant les configurations binaires dans la seconde mémoire et pour la mémorisation d'information de format pour des articles, représentant une table d'indicateurs de données de configurations d'éléments d'image, un dispositif de conversion des données prélevées dans la table classée des articles, la table de données de configurations d'éléments d'image et la table d'indicateurs pour des données de configurations d'éléments d'image en des mots à n bits pour chaque ligne de balayage et au moins une mémoire de colonne d'éléments d'image d'une largeur de n bits, caractérisé en ce que le décodeur comporte une logique de rotation (806) réalisée pour décaler en rotation les mots à n bits dans un registre (827) sur un certain nombre de positions du registre, qui est une fonction de M bits les moins significatifs des informations à m bits, autour de la position des articles sur la page telle que mémorisée dans la première mémoire (503), de manière que $2^M = n$ soit satisfaite, un bloc générateur A0 (805) produisant une adresse de mot dans la mémoire de colonne d'éléments d'image (812, 813) qui est double de celle des m-M bits les plus significatifs desdites informations de position, de manière que les informations qui n'ont pas été décalées en rotation (d—p) du registre (827) de la logique de rotation (806) soient mémorisées dans la mémoire de colonne d'éléments d'image (812, 813) à cette adresse et de manière que les informations du registre (827) qui sont décalées en rotation (a—c) soient mémorisées dans la position d'adresse de mot immédiatement supérieure dans la mémoire de colonne d'éléments d'image, un premier registre à décalage (818) à entrée en parallèle, sortie en série, qui peut être connecté par un tampon (816) a une position d'adresse impaire de la mém ire de colonne d'éléments d'image (812, 813), un second registre à décalage (819) à entrée en parallèl , sortie en série qui, en relation avec le premier registre à décalage (818), peut être connecté à une position d'adresse immédiatement supérieure de la mémoire de colonne d'éléments d'image (812, 813), un compteur destiné à produire des adresses successives pour lire la mémoire de colonne d'éléments d'image (812, 813) et un circuit de combinaison (820) relié aux connexions de sortie en série du premier (818) et du second (819) registres à décalage de manière que les deux registres à décalage (818, 819) soient commandés simultanément par horloge et que des données en série soient combinées dans le circuit de combinaison (820) pour former le débit binaire d'éléments d'image en série vers l'analyseur de sortie à trame.

2. Décodeur selon la revendication 1, caractérisé en ce que la logique de rotation (806) comporte un circuit de décalage en tonneau qui est connecté à un registre d'entrée (825) de n bits, un registre de sortie (827) de n bits et un circuit qui, en fonction de M signaux de commande, peut décaler les données présentes dans le registre d'entrée (825) en rotation et les placer dans le registre de sortie (827).

3. Décodeur selon la revendication 1 ou 2, caractérisé en ce que la mémoire de colonne d'éléments d'image (812, 813) est constituée par n éléments de mémoire dont, à partir de chaque élément de mémoire, une ligne d'adresse est reliée à l'une des n sorties du bloc générateur A0 (805) et dont les autres lignes d'adresse sont reliées ensemble.

4. Décodeur selon les revendications 1 à 3, caractérisé en ce que le bloc générateur A0 (805) comporte M entrées qui sont reliées aux M bits les moins significatifs des informations à m bits concernant la position des articles sur la page et, en réponse aux informations sur les entrées, produisant une configuration binaire pour les n sorties, qui est utilisée pour adresser des éléments de mémoire de la mémoire de colonne d'éléments d'image (812, 813) qui doivent contenir les informations décalées en rotation à une adresse qui est située à une position au-dessus de la position d'adresse indiquée par les m-M bits les plus significatifs de la position Y du mot à n bits qui est en cours de traitement.

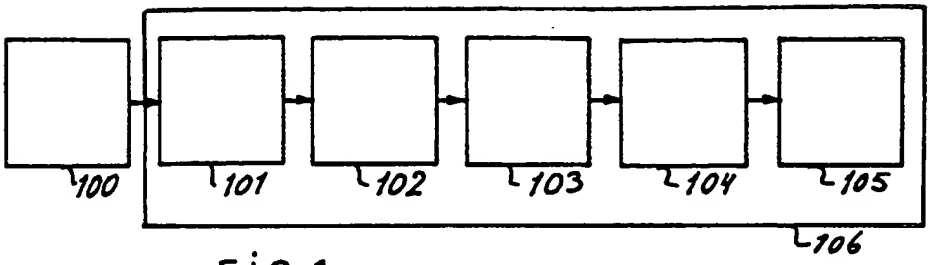

100
101
102
103
104
105
106

FIG. 1

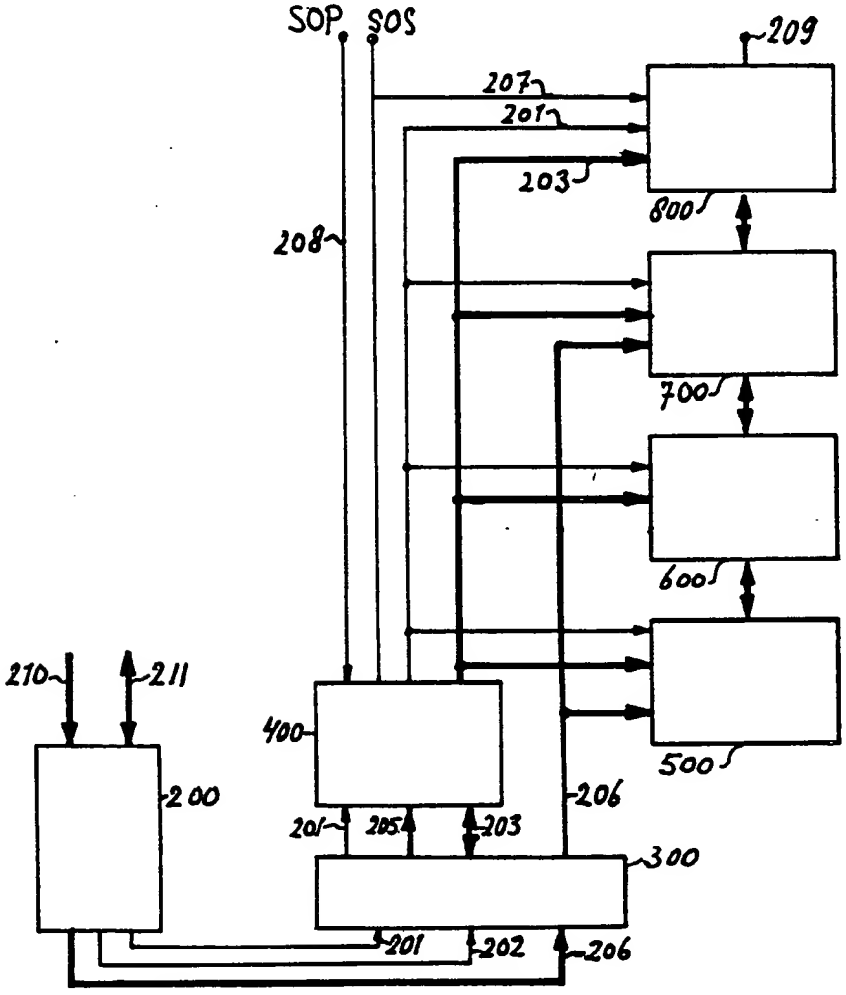

SOP
SOS
209
207
201
203
800
208
700
600
500
210
211
400
200
206
201
205
203
300
201
202
206

FIG. 2

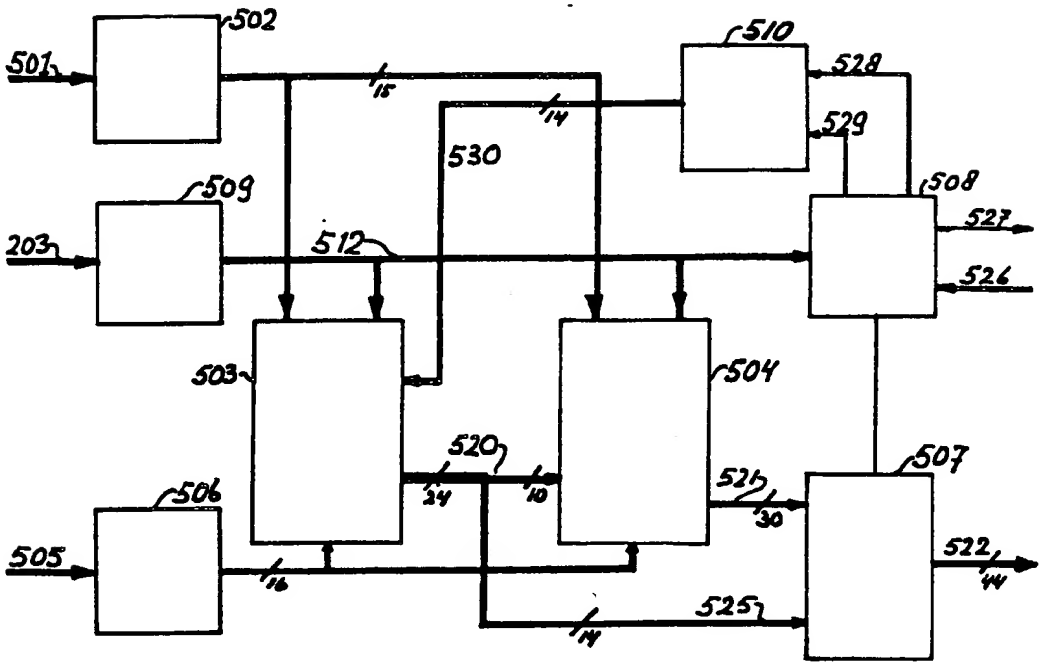

501
502
15
14
510
528
529
530
509
203
512
508
527
526
503
504
520
24
10
507
506
505
16
30
522
44
525
14

FIG. 3

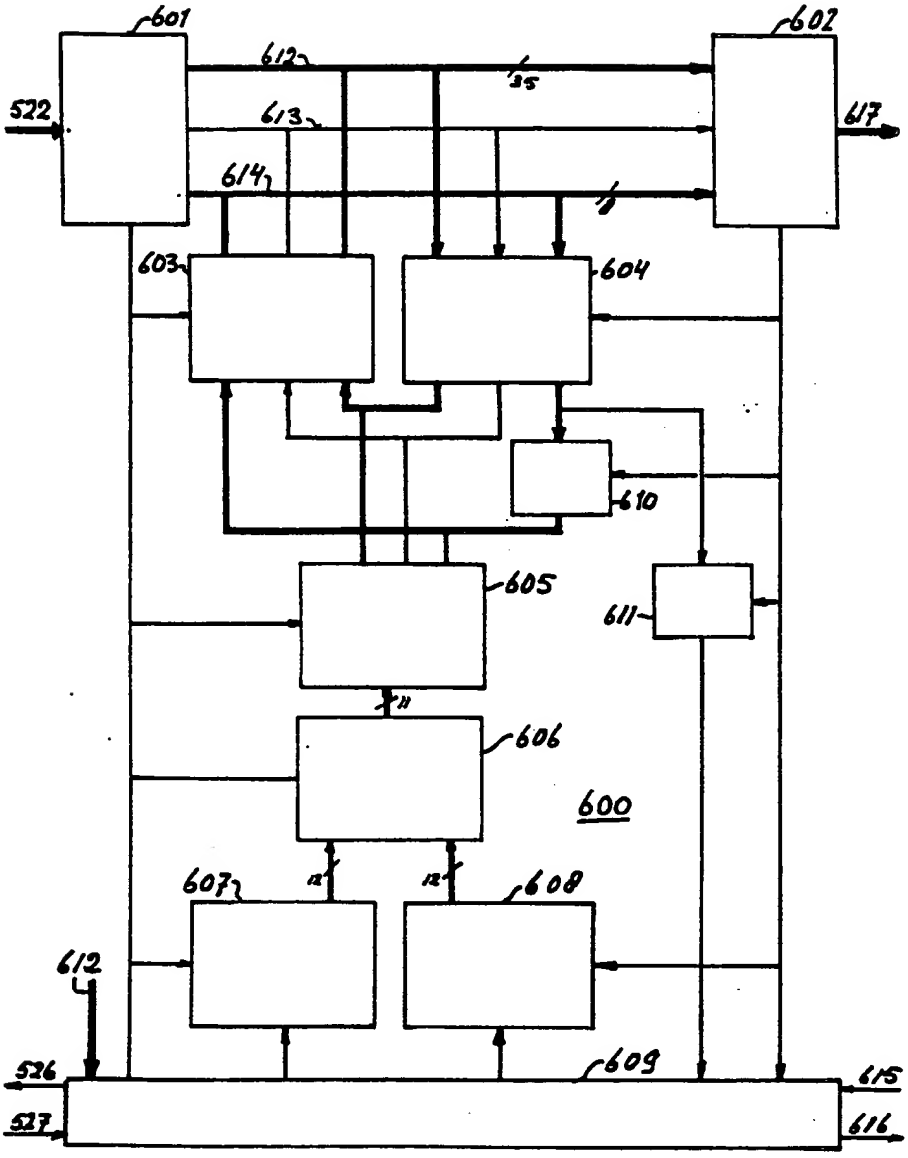

601
602
612
35
522
613
617
614
603
604
610
605
611
11
606
600
607
12
608
612
526
609
615
527
616

FIG. 4

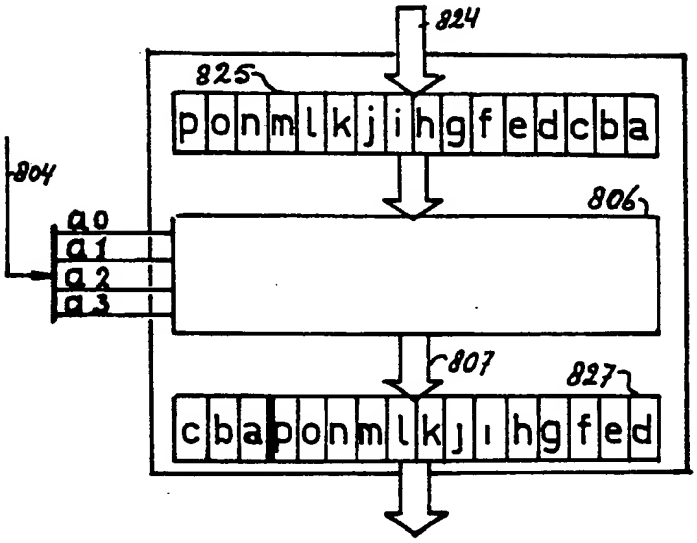

824
825
p o n m l k j i h g f e d c b a
804
806
a 0
a 1
a 2
a 3
807
827
c b a p o n m l k j i h g f e d

FIG. 8

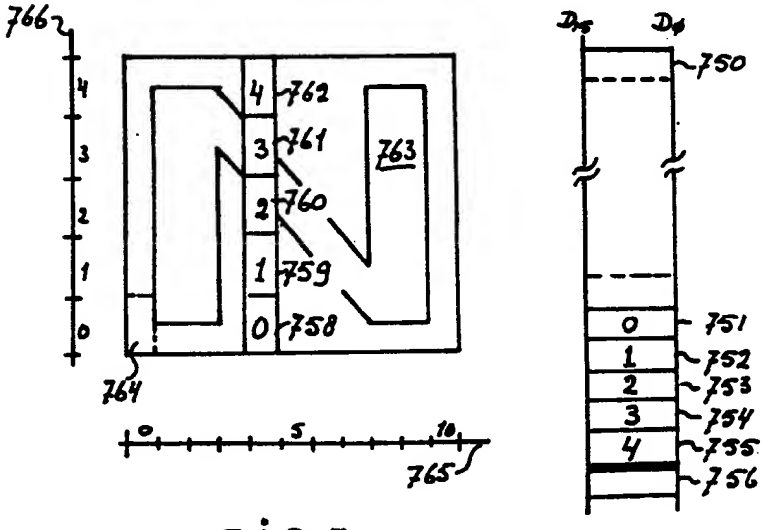

766
4
3
2
1
0
762
761
760
759
758
763
764
0
5
10
765
D5
D0
750
0
1
2
3
4
751
752
753
754
755
756

FIG. 5

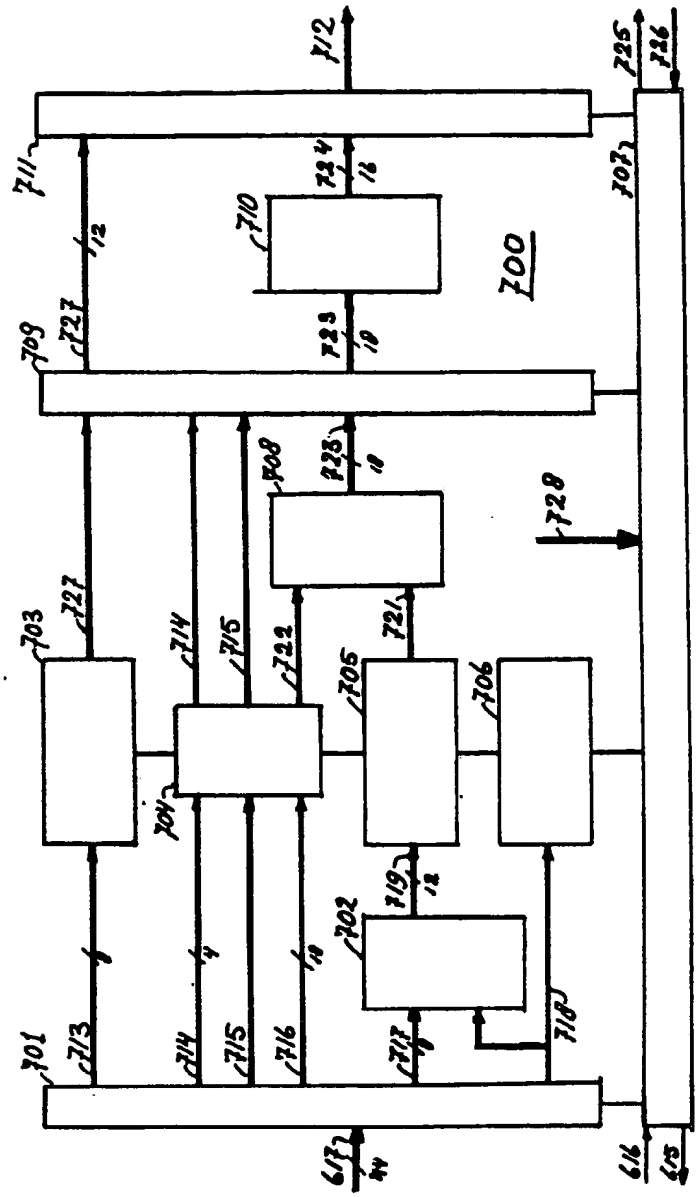

712
725
726
711
724
12
710
700
707
709
727
723
10
708
728
703
721
714
715
722
705
706
704
719
12
702
701
713
714
715
716
717
718
617
44
616
615

FIG. 6

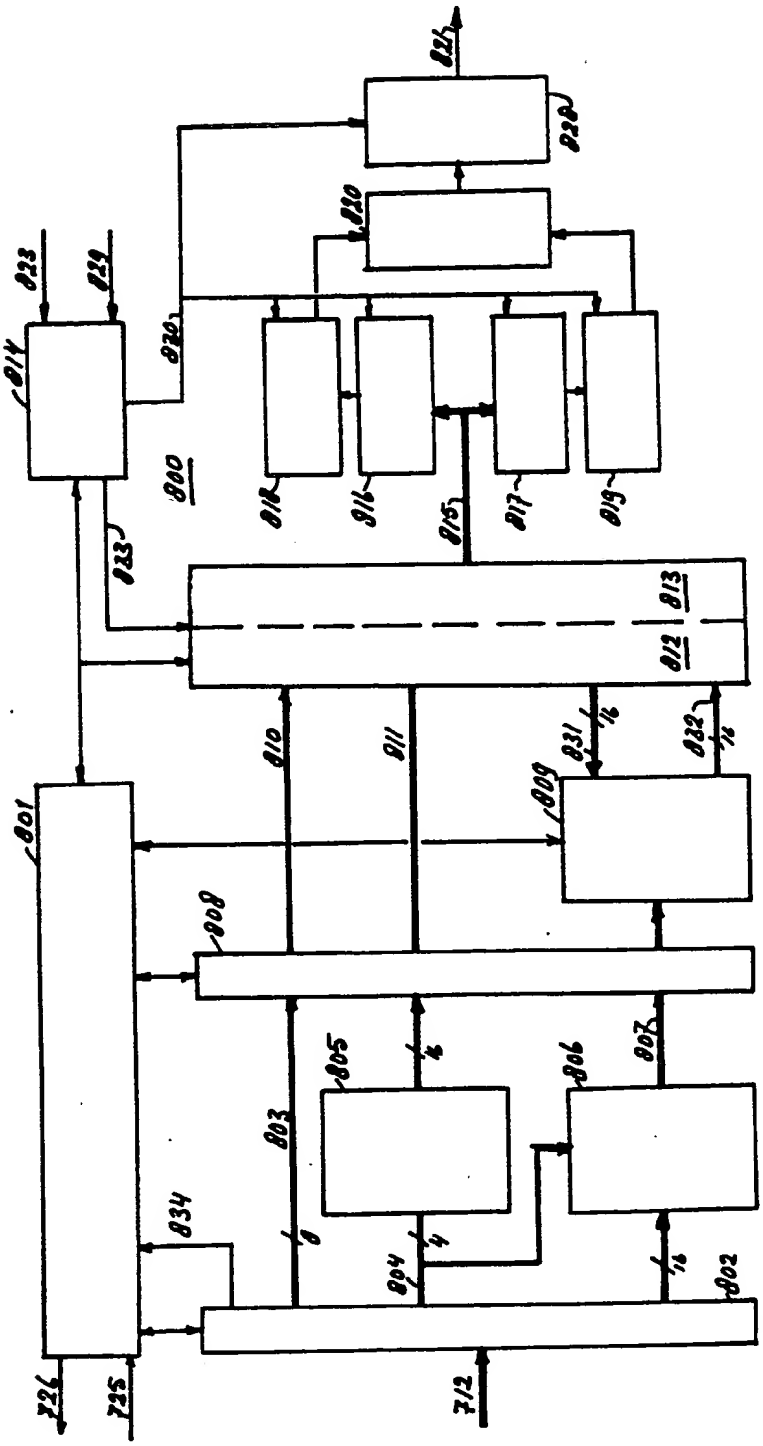

826
820
823
824
814
820
800
818
816
815
817
819
823
813
812
810
811
831
832
809
801
909
805
806
807
803
834
804
802
712
726
725

FIG. 7